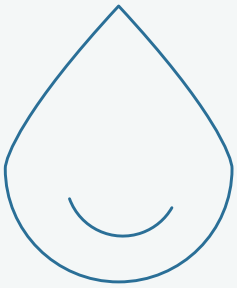
PROBABILISTIC GRAPH MODELING



Probabilistic Modeling of Water Distribution Networks

Leaks, state estimation and where to put the next sensor

386 pages

Ricardo Marquez, Ph.D.

ricardo-marquez.com

Probabilistic Modeling of Water Distribution Networks

Ricardo Marquez

Working draft, September 2026

Preface

This book is written for an engineer who knows conservation laws and linear algebra and has never met a probabilistic graphical model. It takes one idea and follows it through: that the structure of a physical system, the way conserved quantities accumulate at places and move between them and cross the system's boundary, is also the structure of every probabilistic question one can ask about that system. A model of the physics is a graph. A model of what we do not know about the physics is the same graph with probability attached to it. Inference on that graph is how a measurement in one place becomes knowledge about another, how a failure is diagnosed, how a decision is justified.

A water distribution network is a good system to point that idea at. Its physics is settled: a mass balance at every junction and a head-loss law along every pipe, equations engineers have solved by hand and by computer for the better part of a century. So the difficulties the book meets are difficulties of knowing, not of modeling. Its instruments are few: a meter at each district inlet and pumping station, pressure loggers on a handful of hydrants, and thousands of pipes in between that nobody measures. Its parameters age: roughness coefficients were assigned when the pipes were laid and have been changing since. And its most expensive questions are inference questions. Where is the water going that enters a district and never reaches a customer? Which of several places could a burst be? What can the network still deliver with a main out of service?

Every system worth modeling is a piece cut out of a larger one, trading a conserved quantity with an environment it does not model. For a water network the piece is a district, a pressure zone or a whole city. The boundary is where the utility's model stops: the reservoirs and sources that feed it, the service connections that draw from it, and the metered mains that cross into a neighboring district. That boundary is where the modeler's knowledge is thinnest. As the later chapters show, it is also where the mathematics of inference does its most useful work, because the flows and heads at a boundary are exactly what the rest of the network needs to know about the inside. Chapter 1 gives such a system a name, a *boundary-exchanging system*, and the book returns to the boundary at every level.

Nothing in the method of Parts I to V is specific to water. The balances, closures and ports are the same whether the conserved quantity is mass, heat, charge or power, and Table 1.3 lines the four up. This book carries one conserved quantity, mass, from beginning to end. The one exception, the district cooling loop of Example 1.3, puts heat on the same graph to show that nothing in the method assumes a single ledger. Three companion editions take the same method text through electric power transmission, district heating and cooling, and data centers. Each is complete in itself, and a reader of this one needs none of them.

The Introduction says why the book exists, what it inherits from four older traditions, what it adds, and how to read it. Part I is deterministic and establishes the objects. Part II attaches probability to them and states, rather than skips, the places where that attachment needs care.

Part III is inference. Part IV asks the questions an operator actually asks. Part V handles time, hierarchy and scale. Part VI is four complete studies. They run from a loop small enough to check by hand, through the EPA's Net3 and a leak-detection benchmark, to a year of recorded operating data from the L-Town network, scored against the benchmark's published entries. Each gives up one of the modeler's advantages that the one before it had, and each can be reproduced from the material in the earlier parts. Each chapter ends with a short section of notes that says which of its ideas are inherited and from where, and which are the book's own.

Contents

Preface **ii**

0 Introduction **1**
0.1 The questions a water engineer asks . . . 1
0.2 Where the graph comes from . . . 2
0.3 One subject, not four . . . 3
0.4 Probability in the structure, not around the solver . . . 4
0.5 The boundary . . . 6
0.6 From "what is the state" to "what can I know" . . . 6
0.7 What is inherited and what is the book's . . . 7
0.8 How to read this book . . . 7
0.9 What this book does not do . . . 9
Notes and further reading . . . 9

I Physical systems as graphs **10**

1 Conservation, closure, and the boundary **11**
1.1 Three questions about any system . . . 11
1.2 Nodes conserve . . . 12
1.2.1 The balance law . . . 12
1.2.2 Node roles . . . 13
1.2.3 Incidence . . . 13
1.3 Edges close . . . 14
1.3.1 Closure relations . . . 14
1.3.2 A closure is a relation, not an assignment . . . 14
1.3.3 Parameters . . . 15
1.4 The boundary . . . 15
1.4.1 Reservoirs and imposed conditions . . . 15
1.4.2 Exactness at a cut . . . 16
1.4.3 Why the boundary matters most . . . 17
1.5 The residual system . . . 17
1.5.1 Well-posedness . . . 18
1.5.2 Sparsity . . . 18
1.6 Three worked examples . . . 19
1.7 In practice . . . 27

1.8 What is missing . . . 28
Notes and further reading . . . 29
Summary . . . 29
Exercises . . . 30
Solutions to selected exercises . . . 30

2 From equations to factor graphs 32
2.1 The rows as a graph . . . 32
2.2 Three languages . . . 33
2.2.1 Directed graphs . . . 33
2.2.2 The directionality problem . . . 34
2.2.3 Undirected graphs . . . 36
2.2.4 Factor graphs . . . 36
2.3 The physical system as a factor graph . . . 37
2.3.1 A junction is not a variable . . . 38
2.3.2 Two domains on one graph . . . 38
2.4 The factor graph is not unique . . . 39
2.5 Vocabulary . . . 41
2.6 In practice . . . 42
2.7 What is missing . . . 42
Notes and further reading . . . 43
Summary . . . 43
Exercises . . . 44
Solutions to selected exercises . . . 44

II Probability on a physical graph 46

3 Where probability enters 47
3.1 A probability on the state . . . 47
3.2 Five kinds of factor . . . 48
3.2.1 Priors . . . 48
3.2.2 Closure factors . . . 49
3.2.3 Conservation factors . . . 49
3.2.4 Measurement factors . . . 50
3.2.5 Failure factors . . . 50
3.2.6 The complete list . . . 51
3.3 The joint as an objective . . . 51
3.3.1 Residuals in units of their widths . . . 52
3.3.2 Conditioning a Gaussian on one reading . . . 52
3.4 A worked example: one sensor, then two . . . 55
3.5 A second example: the triangle with a flow meter . . . 58
3.6 Reading the graph . . . 60
3.7 In practice . . . 61
3.8 What is missing . . . 62
Notes and further reading . . . 62
Summary . . . 63

Exercises . . . 63
Solutions to selected exercises . . . 64

4 Hard constraints, soft constraints, and the manifold 65
4.1 The hard joint and its manifold . . . 65
4.2 Conditioning on the manifold . . . 66
4.3 The soft joint . . . 67
4.4 Why soften . . . 68
4.5 Choosing the widths . . . 69
4.6 Worked example: the chain with soft physics . . . 71
4.7 A second example: the loop with an unmodeled leak . . . 73
4.8 In practice . . . 76
4.9 What is missing . . . 76
Notes and further reading . . . 77
Summary . . . 77
Exercises . . . 78
Solutions to selected exercises . . . 78

5 Factorization, separators, and treewidth 80
5.1 Factorization is not independence . . . 80
5.2 Reading independence off the graph . . . 81
5.3 Explaining away, precisely . . . 82
5.4 Ports are separators . . . 83
5.5 Elimination and fill . . . 84
5.5.1 One elimination . . . 84
5.5.2 Order and width . . . 85
5.5.3 What this means for a physical graph . . . 86
5.6 Worked examples: counting width and fill . . . 87
5.7 Why separators matter for cost . . . 89
5.8 In practice . . . 91
5.9 What is missing . . . 92
Notes and further reading . . . 92
Summary . . . 92
Exercises . . . 93
Solutions to selected exercises . . . 93

6 The linear-Gaussian case 95
6.1 The Gaussian in information form . . . 95
6.2 The most probable state . . . 96
6.2.1 Gauss–Newton against Newton . . . 97
6.3 Elimination is Cholesky . . . 97
6.3.1 What the factorization costs . . . 98
6.4 The Laplace posterior . . . 100
6.5 Worked example: the chain in matrix form . . . 100
6.6 Second worked example: the water loop in matrix form . . . 103
6.7 Filters, fields, and state estimators . . . 106
6.8 In practice . . . 106

6.9 What is missing . . . 107
Notes and further reading . . . 107
Summary . . . 108
Exercises . . . 108
Solutions to selected exercises . . . 109

7 Beyond Gaussian: hybrid states and nonlinearity 111
7.1 Four ways out of the Gaussian world . . . 111
7.2 Hybrid states: the mixture over branches . . . 112
7.2.1 Worked example: a pipe that may be shut . . . 113
7.2.2 Many discrete variables . . . 115
7.3 Piecewise closures and regimes . . . 115
7.3.1 Worked example: a pressure-reducing valve . . . 116
7.4 Strong nonlinearity . . . 117
7.5 Bounded parameters . . . 119
7.6 What survives . . . 119
7.7 In practice . . . 120
7.8 What is missing . . . 121
Notes and further reading . . . 121
Summary . . . 122
Exercises . . . 122
Solutions to selected exercises . . . 123

III Inference 126

8 Exact inference: elimination, junction trees, Schur complements 127
8.1 What exact means . . . 127
8.2 Messages on a tree . . . 127
8.3 Loops: the junction tree . . . 129
8.4 Regions and ports: the Schur complement as a message . . . 130
8.5 Reuse . . . 132
8.6 Worked example: two districts, one transfer main . . . 133
8.7 Second worked example: two districts and a boundary main . . . 135
8.8 Discrete variables by region . . . 137
8.9 In practice . . . 138
8.10 What is missing . . . 139
Notes and further reading . . . 139
Summary . . . 140
Exercises . . . 140
Solutions to selected exercises . . . 141

9 Approximate inference 143
9.1 When exact is too expensive . . . 143
9.2 Loopy belief propagation . . . 144
9.2.1 On the book's models . . . 144
9.3 Variational methods and expectation propagation . . . 147

9.4 Simulation inside a region . . . 148
9.4.1 What a mixture message loses when collapsed . . . 149
9.4.2 Worked example: a treatment works as a simulated region . . . 149
9.5 Pruning by evidence . . . 151
9.6 The hierarchy, and how to choose . . . 153
9.7 In practice . . . 153
9.8 What is missing . . . 154
Notes and further reading . . . 154
Summary . . . 155
Exercises . . . 155
Solutions to selected exercises . . . 156

10 Enumeration, Monte Carlo, and factorized inference 158
10.1 The question, stated honestly . . . 158
10.2 The problem . . . 159
10.3 Enumeration: the product grid . . . 159
10.4 Monte Carlo . . . 161
10.5 Attribution: where the structural difference appears . . . 162
10.6 The factorized route: cuts of the dependency graph . . . 163
10.7 Second worked example: two works on one river . . . 164
10.8 Beyond three inputs . . . 167
10.9 The claim . . . 169
10.10In practice . . . 169
10.11What is missing . . . 170
Notes and further reading . . . 170
Summary . . . 171
Exercises . . . 171
Solutions to selected exercises . . . 172

11 Sequential evidence 173
11.1 Readings arrive one at a time . . . 173
11.2 One reading, one rank-one update . . . 173
11.3 The innovation . . . 174
11.4 Many readings . . . 175
11.5 Worked example: a stream of flow readings . . . 176
11.6 The Laplace posterior and re-linearization . . . 177
11.7 The value of a reading before it is taken . . . 179
11.8 Cost at scale . . . 180
11.9 Worked example: where to put the next pressure logger . . . 181
11.10In practice . . . 182
11.11What is missing . . . 183
Notes and further reading . . . 183
Summary . . . 184
Exercises . . . 185
Solutions to selected exercises . . . 185

IV The operator's questions 187

12 Conditioning, diagnosis, and virtual sensors 188
12.1 Everything is a readout . . . 188
12.2 Virtual sensors . . . 188
12.3 Is the model adequate? . . . 189
12.4 Where is it wrong? . . . 190
12.5 Worked example: bad data on a water network . . . 191
12.6 Which explanation? . . . 192
12.7 Can this parameter be recovered? . . . 194
12.8 Worked example: can tuberculation be seen? . . . 194
12.9 Which sensor next? . . . 197
12.10Should the Gaussian be believed? . . . 197
12.11In practice . . . 199
12.12What is missing . . . 200
Notes and further reading . . . 200
Summary . . . 200
Exercises . . . 201
Solutions to selected exercises . . . 202

13 Interventions and counterfactuals 203
13.1 Why the graph is not enough . . . 203
13.2 Rows are mechanisms; inputs are exogenous . . . 203
13.3 Seeing versus doing, on the chain . . . 204
13.4 Counterfactuals: three steps . . . 206
13.5 Decisions as branches . . . 209
13.6 Worked example: a drought week . . . 210
13.7 Worked example: holding the transfer flow . . . 211
13.8 What the semantics does not decide . . . 213
13.9 In practice . . . 213
13.10What is missing . . . 214
Notes and further reading . . . 214
Summary . . . 214
Exercises . . . 215
Solutions to selected exercises . . . 216

14 Attribution, sensitivity, and certificates 219
14.1 Three questions, one derivative . . . 219
14.2 Sensitivities from the solve . . . 219
14.3 Attribution . . . 220
14.4 A census with gradients . . . 222
14.4.1 What the census answers . . . 223
14.4.2 The three-main problem . . . 223
14.5 The kink . . . 225
14.6 Worked example: convexity lost and recovered . . . 226
14.7 Conditions . . . 228
14.8 In practice . . . 229

14.9 What is missing . . . 230
Notes and further reading . . . 231
Summary . . . 231
Exercises . . . 232
Solutions to selected exercises . . . 232

15 Combinatorial failure **234**
15.1 The question . . . 234
15.2 The prior over failure sets . . . 235
15.2.1 Independent failures . . . 235
15.2.2 Failure as an intervention . . . 236
15.3 The consequence per branch, and how to get it cheaply . . . 236
15.4 Worked example: seven pipes, one district . . . 237
15.5 The posterior over failure sets . . . 240
15.6 Enumeration by regions . . . 241
15.7 The worst k-set as a query . . . 242
15.8 Common cause and correlated failure . . . 242
15.8.1 A shared shock . . . 243
15.8.2 A mixture over the rate . . . 243
15.9 Worked example: a booster station . . . 243
15.10In practice . . . 245
15.11What is missing . . . 246
Notes and further reading . . . 246
Summary . . . 247
Exercises . . . 247
Solutions to selected exercises . . . 248

V Time, hierarchy, and scale **249**

16 Dynamics and temporal factors **250**
16.1 The accumulation term made live . . . 250
16.2 Integration as a sequence of steady solves . . . 251
16.3 The unrolled graph . . . 252
16.4 Process noise . . . 253
16.5 Filtering, smoothing, and prediction as eliminations . . . 254
16.6 Worked example: tracking a drifting source . . . 255
16.7 Detecting a change in time . . . 256
16.8 Two hypotheses the steady state cannot separate . . . 257
16.9 Events and cascades . . . 258
16.10Worked example: two service reservoirs . . . 259
16.11In practice . . . 261
16.12What is missing . . . 262
Notes and further reading . . . 262
Summary . . . 263
Exercises . . . 263
Solutions to selected exercises . . . 264

17 Hierarchy as physical decomposition **265**
17.1 One conservation law used twice . . . 265
17.2 Nested messages . . . 266
17.3 What the parent needs to know . . . 267
17.4 Worked example: a supply zone in three levels . . . 268
17.5 The cut, three times . . . 272
17.6 Worked example: a metered zone under a junction . . . 273
17.7 Hierarchy in time . . . 275
17.8 In practice . . . 277
17.9 What is missing . . . 277
Notes and further reading . . . 278
Summary . . . 278
Exercises . . . 278
Solutions to selected exercises . . . 279

18 Partitioning at scale **281**
18.1 The cost of a cut . . . 281
18.2 Elimination width of physical networks . . . 282
18.3 Choosing the cut . . . 283
18.4 Coupling the regions: messages, propagation, or iteration . . . 285
18.5 Enumeration by region . . . 289
18.6 A second domain, on purpose . . . 292
18.7 In practice . . . 293
18.8 What is missing . . . 293
Notes and further reading . . . 293
Summary . . . 294
Exercises . . . 294
Solutions to selected exercises . . . 295

VI Case studies **296**

19 Case studies **297**
19.1 What a case study is here . . . 297
19.2 Each chapter removes an advantage . . . 298
19.3 What is not here . . . 298
19.4 Reading a case study . . . 298

20 Three questions on one set of rows **300**
20.1 The question . . . 300
20.2 The system . . . 301
20.2.1 Why a closed form exists . . . 301
20.2.2 One regularization, and what it costs . . . 302
20.3 The model as a factor graph . . . 302
20.4 Method . . . 303
20.5 Results . . . 304
20.5.1 The state, four ways . . . 304

20.5.2 How big, and whether the interval is honest . . . 305
20.5.3 Which junction . . . 306
20.5.4 Where two gauges stop seeing . . . 306
20.5.5 Cost . . . 307
20.6 What surprised . . . 308
20.7 Reproduction . . . 308
Notes . . . 309

21 Half the pipes have meters **310**
21.1 The question . . . 310
21.2 The system . . . 311
21.3 The model as a factor graph . . . 311
21.4 Method . . . 311
21.5 Results . . . 313
21.5.1 Is the error bar honest? . . . 313
21.5.2 It is weighted least squares . . . 314
21.5.3 What the cut separates . . . 314
21.5.4 Where are the breaks? . . . 316
21.5.5 Two hypotheses that converge slowly . . . 317
21.5.6 Cost . . . 318
21.6 What surprised . . . 318
21.7 Reproduction . . . 319
Notes . . . 319

22 Screen, then confirm **320**
22.1 The question . . . 320
22.2 The two networks . . . 320
22.3 The model: one unknown, many snapshots . . . 321
22.4 Method . . . 322
22.5 Results . . . 323
22.5.1 Which junction leaks . . . 323
22.5.2 What a window buys . . . 323
22.5.3 What the leak competes with . . . 324
22.5.4 A wrong roughness becomes a leak . . . 325
22.5.5 Modena: screen, then confirm . . . 326
22.5.6 Cost . . . 327
22.6 What surprised . . . 327
22.7 Reproduction . . . 328
Notes . . . 328

23 Held to what a utility knows **330**
23.1 The question . . . 330
23.2 The system . . . 331
23.3 The two layers . . . 331
23.4 Method . . . 331
23.5 Results . . . 332
23.5.1 What calibration buys, and what it does not . . . 332

23.5.2 The year, read forward . . . 333
23.5.3 The official score . . . 334
23.5.4 Are the stated probabilities honest? . . . 334
23.5.5 The second evaluation, which is worse . . . 335
23.5.6 Cost . . . 336
23.6 What surprised . . . 336
23.7 Reproduction . . . 336
Notes . . . 337

A Notation **338**
A.1 Conventions . . . 338
A.2 Symbols . . . 339
A.3 Letters with more than one meaning . . . 342

B Gaussian identities **344**
B.1 The two forms of a Gaussian . . . 344
B.2 Block matrices . . . 345
B.3 Marginals and conditionals . . . 346
B.4 Products and linear-Gaussian models . . . 347
B.5 Rank-one changes . . . 348
B.6 Integrals, quadratic forms and divergences . . . 348
B.7 The Laplace approximation . . . 349
B.8 Where the book uses each identity . . . 350

C From the book to the engine **351**
C.1 What TERRA is . . . 351
C.2 A model in a dozen lines . . . 352
C.3 Where each construction lives . . . 353
C.4 The case studies . . . 353

D A catalogue of closures **356**
D.1 How to read an entry . . . 356
D.2 The four patterns in the engine . . . 358
D.3 Thermal . . . 358
D.4 Fluid . . . 360
D.5 Moist air . . . 362
D.6 Electrical . . . 362
D.7 Storage and electrochemistry . . . 364
D.8 Plant equipment . . . 365

Chapter 0

Introduction

In the small hours, when almost nobody is drawing water, a district metered area tells the truth about itself. The meter at its inlet reads what enters; the customers are asleep; what the district still takes is, nearly all of it, what its pipes are losing. When the night flow climbs, the control room has a short list of explanations and a short list of instruments with which to choose between them. A main has burst somewhere along a few dozen kilometres of pipe. Or the inlet meter has drifted. Or a boundary valve that the records say is shut is passing water in from the neighboring district, and the district is not losing more water at all, only being metered wrongly. Or a large customer is filling a tank. The instruments are the inlet meter, a handful of pressure loggers on hydrants, and a hydraulic model last calibrated against a fire-flow test some years ago.

The engineer on shift already has the physics of that district: a balance at every junction, a head-loss relation along every pipe, a fixed head at every reservoir and every boundary. Given the demands, a solver turns those equations into every head and every flow. What the solver cannot do is run the other way: take three pressure readings and one flow reading and say which of the four explanations they support, how strongly, along which stretch of main a crew should start listening, and whether one more logger would settle it. Probabilistic graphical models supply that machinery. They are usually taught apart from conservation laws, head-loss curves and network solvers, and the engineer who wants both has had to learn two subjects and build the bridge alone. This book is the bridge. Its claim is that the bridge is short, because the graph the probabilistic machinery needs is already there in the network equations the engineer wrote, and that once the graph is recognized, every question on the shift's list is a readout of one object. This chapter says why that claim is worth a book, what the book inherits from four older traditions, what it adds, and how to read it.

0.1 The questions a water engineer asks

A model of a physical system, as engineering teaches it, is a set of equations and a solver. The equations are the balances, the closures and the boundary conditions; the solver turns known inputs into a computed state. The pipeline is

$$\text{inputs} \longrightarrow \text{solver} \longrightarrow \text{outputs}, \tag{0.1}$$

and it answers one question well: given these inputs, what is the state? Most textbooks on physical modeling stop there, and for design that is often enough.

Operation asks different questions. A water network is instrumented, and its instruments are sparse: meters at the district boundaries and the pumping stations, pressure loggers on a few chosen hydrants, almost nothing in between. They are noisy, and occasionally wrong. The pipes' roughness coefficients were assigned when the pipes were laid and have been changing since. The demands are forecasts. The valves are open or shut on a drawing that may be out of date. The questions an operator or a network engineer actually asks are:

- What are the heads and flows across the district now, given a few pressure readings and a model I only partly trust?
- Which pipe's roughness is off, and by how much, and is it the model that is wrong or the network?
- What is the pressure at the critical node nobody logs, the high point where customers complain first?
- Is the night flow a burst, a drifting meter, a passing boundary valve or legitimate use, and if it is a burst, on which stretch of main?
- How uncertain is each of those answers?
- Where should the next pressure logger go, and how much would it narrow the search?
- What would happen if I shut that valve, slowed that pump, or lowered that pressure-reducing valve's setpoint?
- Which uncertain demand or roughness drives the risk of low pressure at peak, and is the number I am about to report a floor, a ceiling or a guess?

None of these is a question the pipeline (0.1) can answer. Each asks the model to run in a direction other than inputs to outputs, or to weigh a reading against a law, or to compare explanations, or to say how far its own answer can be trusted. The pipeline has no place to put a reading, no notion of an explanation, and no measure of trust.

The book's starting observation is that the equations do not have to be rewritten to answer these questions. They have to be read differently. A balance says that certain flows sum to zero at a node; a closure says that a flow and two potentials satisfy a relation; a boundary condition says that a quantity has a value. Each is a statement about which combinations of the variables are admissible. Attach to each a statement of how far from admissible the system may be, attach to each uncertain input a statement of what is known about it, attach to each sensor a statement of what it read and how well, and the collection is a probability distribution over the entire state of the system, built from nothing but the physics and the instruments. Every question in the list is then a question about that distribution, and Parts II to IV of this book are about answering them.

0.2 Where the graph comes from

A textbook on probabilistic graphical models begins with random variables and their joint distributions, defines directed and undirected graphs and the factorizations they encode, develops conditional independence, and then teaches inference on the graph. The examples are alarms and burglaries, diseases and symptoms, letters and their noisy transmissions. It is a coherent subject and the machinery is powerful, but an engineer arriving from the physical side asks a question the textbook does not answer: where did the graph come from? For an alarm and a burglary the graph is a modeling decision, drawn from a story about what causes what. For a pumping main, a district metered area or a city's trunk network it is not a decision at all.

For a physical model the graph is the sparsity pattern of the equations. Each balance reads

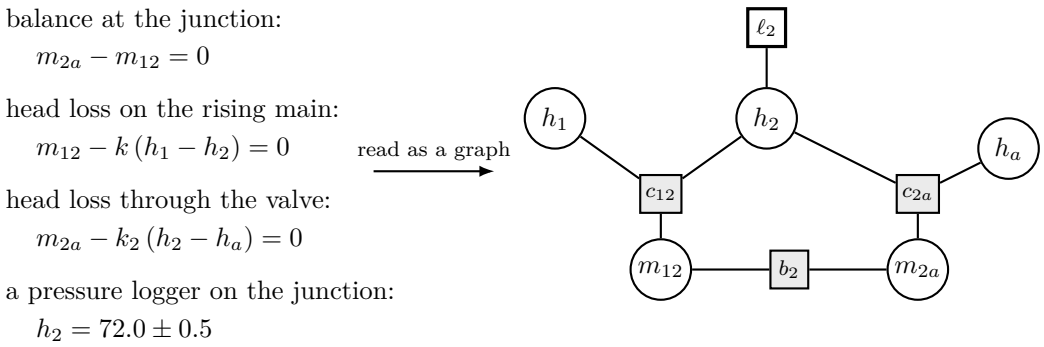


Figure 0.1: The graph comes from the equations. Left: three physics equations of the pumping chain of Example 1.1 and one pressure-logger reading, as an engineer writes them. Right: the same four statements as a factor graph. Each equation is a square joined to exactly the variables it reads; the sensor is a square of degree one. Nothing was decided in drawing it.

Table 0.1: What each object of a physical model becomes when probability is attached. The graph is unchanged; only the meaning of the squares is.

In the equations	In the factor graph	With probability attached
balance at a node	a factor on the node's flows	conservation, to within a small width
constitutive relation	a factor on a flow and two potentials	closure, to within its model error
boundary condition	a factor of degree one	a prior on the boundary value
uncertain parameter	a variable in every closure reading it	a prior on the parameter
sensor	(absent)	a likelihood of degree one on its variable
component in or out of service	(absent)	a discrete variable in its closure's scope

the flows at one node. Each closure reads one flow and the potentials at its ends. Each boundary condition reads one variable. Draw a node for every variable and a node for every equation, join each equation to the variables it reads, and the result is a factor graph, the bipartite graph on which the probabilistic machinery operates (Figure 0.1). No modeling decision was made in drawing it, because every edge was already an entry of the Jacobian that the solver assembles. The graph is the model's structure, seen.

Attaching probability then changes what the squares mean and not where they are. A balance equation becomes a factor that says the balance holds to within a small tolerance. A closure becomes a factor that says the constitutive law holds to within its model error. An uncertain parameter or boundary condition gets a prior, a square of degree one on its variable. A sensor becomes a likelihood, another square of degree one, on the variable it reads. A component that may be in service or out becomes a discrete variable in the scope of the closure it governs. Table 0.1 lists the correspondence, which Chapter 3 develops. The bridge from the engineer's equations to the graphical model is this table, and it is short.

0.3 One subject, not four

Little in this construction is new in isolation, and it would be wrong to present it as if it were. Four traditions have each built part of it, and the book is written in the conviction that they are one subject that has been taught as four.

Port-based physical modeling. Bond graphs, port-Hamiltonian systems and network thermodynamics supply the domain-independent view of a physical system as conserved quantities at nodes, flows driven by potentials along edges, and conjugate flow-and-potential pairs at ports, the same structure whether the domain is hydraulic, thermal, electrical or mechanical [39, 66, 69, 95]. This tradition stops at the equations: it says how to write $\mathbf{F}(\mathbf{z}) = \mathbf{0}$ for any physical system, and then solves.

Equation-based modeling. Acausal modeling languages and the structural analysis of differential-algebraic systems establish that a physical law is a relation among variables, not an assignment of an output from inputs, and they give the bipartite equation-variable structure and its matching theory [5, 20, 63, 68]. This tradition, too, stops at the solve, and it does not attach probability.

Probabilistic graphical models and sparse elimination. Factor graphs, Markov properties, separators, elimination, fill and treewidth come from the graphical-model and sparse-matrix literatures [16, 45, 47, 49, 78]. This tradition starts from a factorized distribution and asks how to compute with it; it does not say where the factorization comes from, and its graphs are drawn, not derived.

Bayesian inverse problems and state estimation. Priors on uncertain physical inputs, sensors as likelihoods, weighted least-squares estimation under conservation constraints, and the diagnosis of bad data are established in inverse problems, in power-system state estimation, and in process data reconciliation [1, 37, 57, 64, 88], and in water distribution they are the calibration of network models and the localization of leaks from pressure residuals [71, 79]. This tradition does attach probability to physics, but usually around the solver: the physics is a black-box map from inputs to observables, and the posterior is over the inputs alone.

Put the four end to end and they make one chain,

$$\boxed{\text{physical topology} \;\rightarrow\; \text{local equations} \;\rightarrow\; \text{factorization} \;\rightarrow\; \text{probability} \;\rightarrow\; \text{inference} \;\rightarrow\; \text{engineering decisions}} \tag{0.2}$$

of which each tradition owns one or two links and none carries the whole. Figure 0.2 draws it, with the chapters of this book placed along it. The book's purpose is to make the chain feel like one subject: to start where the engineer starts, at the topology and the equations, and to arrive where the operator needs to be, at a decision with its uncertainty stated, without changing the object along the way.

0.4 Probability in the structure, not around the solver

There is an obvious way to combine a physical model with probability, and most of the fourth tradition uses it: leave the solver as it is, treat it as a function from uncertain inputs to observable outputs, draw inputs from their priors, run the solver, and reason about the outputs. The physics is a black box inside a probabilistic wrapper. Figure 0.3 draws it on the left. It is correct, it is general, and it is how a great deal of uncertainty quantification is done.

The book takes the other route, drawn on the right: the equations themselves are the factors, and probability lives inside the structure, on every variable the physics has. The thesis of the book, in one sentence, is

$$\boxed{\text{Do not put probability around the solver. Put it into the structure of the physical model.}} \tag{0.3}$$

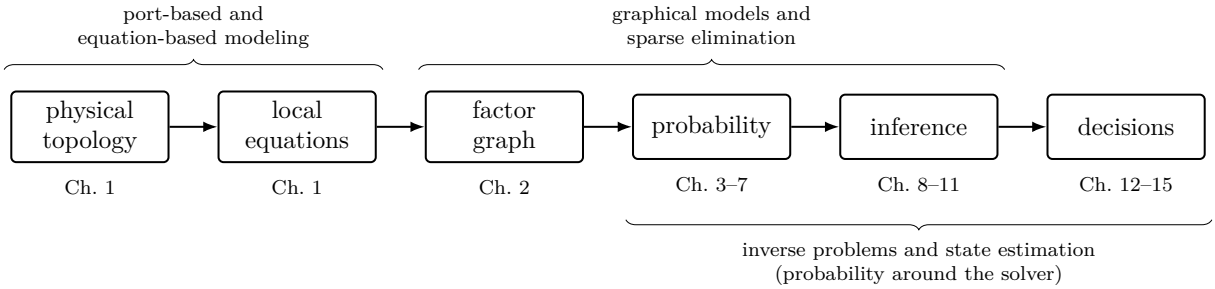
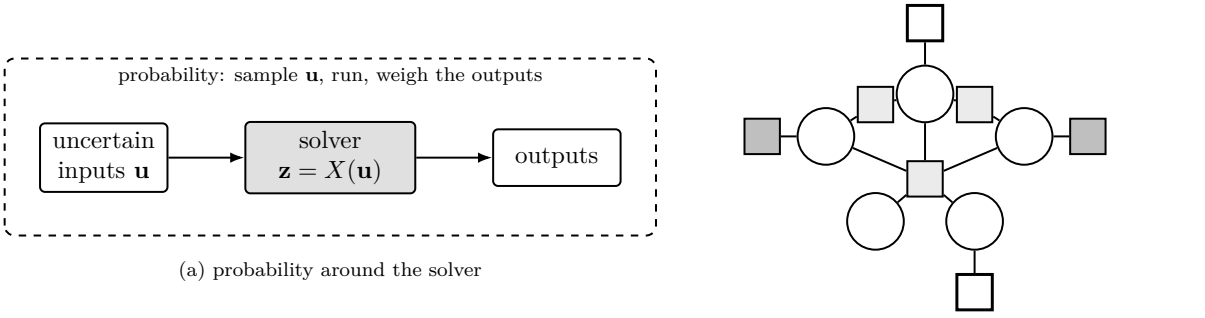


Figure 0.2: The chain of ideas the book follows, with the chapters that cover each link and the traditions that own parts of it. No one tradition carries the object from the topology to the decision.



(a) probability around the solver

(b) probability in the structure:
priors (dark) and sensors (open) added to the physics graph

Figure 0.3: Two ways to combine physics and probability. (a) The solver is a black box; probability lives outside it, over its inputs and outputs. (b) The physics equations are the factors; priors and sensors are squares added to the same graph. The book takes the second route and says why.

Three consequences follow, and they are what the later chapters are about.

There is one object. The deterministic solve is the special case of the probabilistic model in which the priors and sensors are absent and the physics is exact; the two share every row and every Jacobian (Chapter 6 proves this). Adding a sensor adds a row. Adding a question adds nothing: the estimate, the diagnosis, the counterfactual and the risk are readouts of one posterior (Chapter 12).

Structure is preserved, and structure is what computation costs. The wrapper's map from inputs to outputs is dense: every output depends on every input. The physics graph is sparse: every equation reads a few variables. The posterior built in the structure has the sparsity of the physics, and every computation in Part III is affordable because of it. The wrapper's route discards the sparsity the moment it closes the box.

Violation can be seen. A wrapper enforces the physics exactly, because the solver does. When the sensors report something no state of the physics can produce, the wrapper has no answer. The structural model gives each law a width, lets it be violated at a cost, and reports the violation as a diagnosis: the burst, the drifting meter, the valve believed shut (Chapters 4 and 12).

0.5 The boundary

Every system worth modeling is a piece cut out of a larger one. A pressure zone trades water with the zones above and below it, a district metered area trades water with the trunk main that feeds it, a pumping station trades water with its aquifer and head with the network. The piece has an inside that the modeler describes and a boundary across which water passes to an outside the modeler does not describe. Chapter 1 calls such a piece a *boundary-exchanging system*, and the book returns to the boundary at every level for a reason that is easiest to state in the language of the third tradition.

In a graphical-model text, a *separator* is a set of variables whose knowledge makes two parts of the graph independent, and finding one is a graph-theoretic exercise whose reward is a cheaper computation. Water engineers drew their separators before any theory asked them to. A district metered area is defined by its boundary: every main that crosses it is either metered or valved shut, so that what enters is known and a night-flow balance can be struck. The flows and heads at those metered inlets, a flow and its conjugate potential at each port, are exactly the variables that separate the district's interior from the rest of the network (Chapter 5 proves it, and proves that nothing smaller does). So the instruction "find a separator" becomes "cut the network where the meters are", which is where the utility has already cut it, and the mathematics then explains why that cut is also the cheap place to divide the computation. It explains, too, why a boundary valve believed shut and in fact passing water does the damage it does: it is a hole in the separator, and Chapter 8 measures what it costs.

This gives one decomposition three readings at once. Physically, a subsystem is summarized to its neighbors by what crosses its ports, exactly, because conservation says the cut flows equal the net source inside. Probabilistically, the ports are a separator, so a subsystem's message to the rest of the system is a function of the ports alone. Computationally, eliminating the interior onto the ports is the Schur complement of sparse elimination, the Kron reduction of network theory, and the region message of a junction tree, which are one operation (Chapter 8). A district can be an equivalent inlet with error bars, computed once and handed to its neighbors; a large network can be a tree of districts (Chapter 17 works one, a district metered area under a distribution junction); and a partition can be judged by its ports (Chapter 18). The three-way connection is the part of the book for which the author has found the least direct precedent, and it is the reason the boundary is in the title.

0.6 From "what is the state" to "what can I know"

The pipeline (0.1) has a direction: inputs in, outputs out. A physical law does not. The relation

$$m - k_2\,(h_2 - h_a) \;=\; 0 \tag{0.4}$$

says that four quantities satisfy one constraint: the flow through the pumping chain's throttling valve, the heads either side of it, and the valve's conductance. It does not say that m is computed from the heads; given m and h_2 it determines the trunk main's head h_a, and given all three it determines k_2, which is how a valve that is throttled harder than the records say is found. Which variable is unknown is a property of the question, not of the physics (Chapter 2 makes this precise, and shows that for a looped system no fixed direction exists at all). The book keeps the relations as relations, which is what the second tradition insists on and what the first tradition's bond graphs draw, and it is why the factor graph rather than the directed graph is the book's language.

Once uncertainty and observations enter, the absence of direction becomes the point. A reading of one head is evidence about everything the physics ties to it. On the pumping chain that runs through the book, a single pressure logger on the junction cuts the uncertainty in the pumping station's unmeasured delivery by a factor of three and a half, and predicts the unmeasured discharge head to within about a metre and a half, before any second sensor is read (Chapter 3). That prediction is a *virtual sensor*, and its error bar is part of the answer. The question the model answers has changed from "what is the state, given the inputs" to "what can I know, given what I have", and the second question is the operator's.

0.7 What is inherited and what is the book's

A book that combines old ingredients owes its reader a clear account of which is which. Each chapter closes with a section of notes that does this for its own contents. The account for the whole is short.

Inherited: the port-based view of physics; the acausal reading of its equations; the factor graph, its Markov properties, elimination and treewidth; the Gaussian machinery of least squares, sparse Cholesky and the Laplace approximation; Bayesian inverse problems; state estimation and data reconciliation with their bad-data statistics; the causal calculus of interventions; variance-based attribution; reliability evaluation and contingency analysis. None of these is the book's, and the notes say whose each is.

The book's: the synthesis. Taking an equation-based, conservation-structured physical model and turning its residual structure, unchanged, into the factorization on which inference, diagnosis, uncertainty propagation, attribution and decision queries are performed; the identification of physical ports with probabilistic separators and computational cuts; the intervention semantics for acausal equations; and the insistence, carried through every chapter, that the deterministic model is a corner of the probabilistic one.

The author has looked for this treatment and not found it. Pieces of it exist in several communities that have discovered the connection independently. Multibody dynamics has been written as a factor graph whose sparse structure serves both simulation and estimation, with forward and inverse dynamics as different queries on one graph [10], which is the "relations, not assignments" reading in a single domain. Bond graphs have been converted into Bayesian networks for fault diagnosis [54], which attaches probability to a port-based physical model but does so by assigning the causal directions this book declines to assign. Power systems have been estimated by message passing on a factor graph built from the power-flow equations [14], which is Chapters 6 to 12 in one domain and one query. Table 0.2 sets these beside the four traditions and beside this book. What the table shows is not that any column is empty but that no prior row fills them all, and that several communities have arrived at parts of one construction without a text that says they are parts of one construction. That is the gap the book is written to fill, and "not found" is the strongest claim the author is willing to make for it.

0.8 How to read this book

Who it is for. The reader is an engineer who knows conservation laws and linear algebra and has never met a probabilistic graphical model. Nothing about probability beyond the meaning of a density is assumed; Chapter 3 supplies what is needed in a page and Appendix B the Gaussian identities. A reader who knows graphical models and wants the physics will find Chapters 1, 2

Table 0.2: The closest prior work, by what it covers. A check marks a substantial treatment; a word marks a partial or specialized one.

	physics graph	equations as factors	probability	general domains	ports as separators	broad inference
bond graphs, port-Hamiltonian [69, 95]	✓	—	—	✓	✓	—
multibody factor graphs [10]	✓	✓	limited	multibody	—	solve, estimate
bond-graph Bayesian networks [54]	✓	indirect	✓	general idea	—	fault diagnosis
factor-graph state estimation [14]	power grid	specialized	✓	power only	regional	state estimation
graphical-model texts [45, 49]	—	—	✓	abstract	graph-theoretic	✓
inverse-problem texts [37, 88]	equations	not generally	✓	✓	—	inverse only
this book	✓	✓	✓	✓	✓	✓

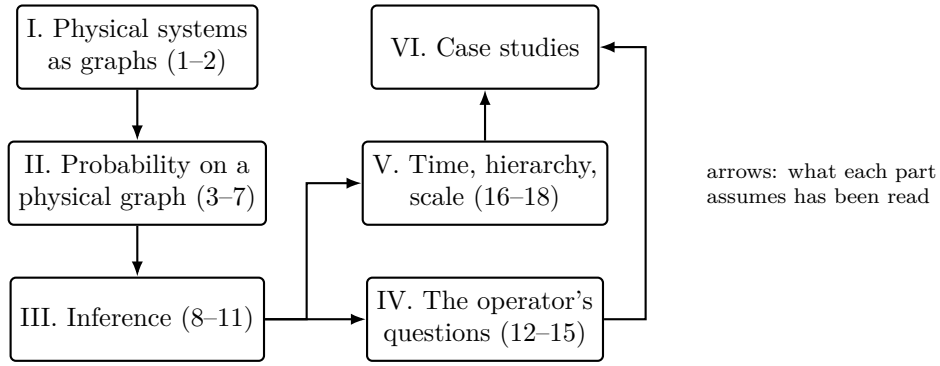


Figure 0.4: The parts of the book and their dependencies. Parts IV and V each need Part III and not each other; Part VI needs both.

and 5 the ones to read first, since they say what the graph is and why the ports separate. The examples are water examples; the constructions are not, and a reader who works on heat, charge, power or any other conserved quantity will find nothing in Parts I to V that needs changing but the names.

The parts. Figure 0.4 shows the six parts and what each depends on. Part I is deterministic and builds the objects. Part II attaches probability and states, rather than skips, the places where that attachment needs care. Part III organizes the computation. Part IV asks the operator's questions. Part V handles time, hierarchy and scale. Part VI is a set of complete studies, each reproducible from the material in the earlier parts.

The running examples. Two small systems appear in almost every chapter: the pumping chain of Example 1.1, a borehole station lifting into a trunk main through two sections of rising main and a throttling valve, and the water loop of Example 1.2, three junctions in a triangle fed from a trunk main, with the head loss linearized so that every number can be checked by hand; Chapter 6 restores the Hazen–Williams exponent and measures what the linearization cost. Both are introduced in Chapter 1, both are small enough to solve by hand, and both acquire priors, loggers, a failed pipe, a second district, an intervention and a failure set as the chapters go on. The reader who follows them through will have seen every construction of the book on a system whose every number can be checked. A third, the district cooling loop of Example 1.3, carries mass and energy on one graph and is kept to show that nothing in the method assumes a

single conserved quantity. Larger systems appear where the small ones cannot show the point: a two-district network joined by one transfer main for decomposition and failure, a booster station with four pumps for common cause, a grid of streets for cost at scale, and the case studies of Part VI.

The numbers. Every number in every worked example is produced by a short script that accompanies the book, one per chapter, and the text says which. The scripts are a few hundred lines each and use nothing beyond a numerical library; they are meant to be read and rerun, and several exercises ask for exactly that. The case studies of Part VI read their numbers from committed result files of the studies they describe. No number in the book was typed by hand.

Conventions. Each chapter ends with a section on what is in practice, notes on what is inherited and from where, a summary, five exercises, and worked solutions to two of them. Load-bearing conclusions are set apart as numbered principles; there are about three per chapter, and a reader who remembers only those will have the book's argument. Propositions are proved where they are stated. Notation is collected in Appendix A, and the reference implementation that the author used to run every script is mapped to the book's constructions in Appendix C; nothing in the body depends on it.

0.9 What this book does not do

It does not teach physical modeling; it assumes the reader can write a balance and a closure, and it uses the simplest ones. It does not teach probability or graphical models in generality; it teaches the parts a physical model needs, and points to the texts that teach the rest. It does not develop machine learning: no factor in this book is learned from data, though Chapter 12 estimates parameters and Chapter 14 fits a surrogate, and the notes say where the learned versions of its constructions live. It does not treat water quality: no chlorine decay, no age, no contaminant tracking, though each would ride on the same graph. It does not treat transients: there is no water hammer here, and time enters in Chapter 16 only as service reservoirs filling and draining. It does not treat continuous space: every system here is a graph of lumped elements, and a field discretized on a mesh appears only as a large graph whose treewidth is a warning. And it does not claim that the constructions it teaches are new; it claims that they are one subject, and it tries to teach them as one.

Notes and further reading

The five gaps that §0.1 to §0.6 describe were articulated in a review of the manuscript, and the positioning of §0.7 follows a literature search made in response to it. The closest works found, Blanco-Claraco, Leanza and Reina's factor-graph multibody dynamics [10], Lo, Wong and Rad's bond-graph Bayesian networks [54], and Chavali and Nehorai's factor-graph state estimation [14], are each discussed in the chapter whose subject they touch. The four traditions of §0.3 are cited there and again in the notes of the chapters that inherit from them; the reader who wants one text per tradition is directed to Karnopp and Rosenberg [39], Benveniste, Caillaud and Malandain [5], Koller and Friedman [45], and Kaipio and Somersalo [37].

Part I

Physical systems as graphs

Chapter 1

Conservation, closure, and the boundary

Every physical system a modeler cares about is a piece cut out of a larger one. A building trades heat with the weather. A transmission network trades power with the networks around it. A district metered area trades water with the zone around it. The piece has an inside, which the modeler describes in detail, and a boundary, across which conserved quantities pass to an outside the modeler does not describe at all.

This chapter builds the deterministic model of such a piece. It has three ingredients and no probability. The three ingredients are conservation at places, closure along paths, and exchange at the boundary. The reason to build the deterministic model first, and to build it carefully, is that the probabilistic model of Part II inherits its structure entirely: the places, paths and boundary of this chapter become the nodes, factors and separators of the graphical model. Nothing is added to the structure later. Only uncertainty is.

1.1 Three questions about any system

Take any conserved quantity: energy, mass, electric charge, a chemical species. Three questions describe how it lives in a system.

1. Where can it accumulate?
2. How does it move from one place to another?
3. What crosses the boundary?

The first question identifies the *places*. Each place has an amount of the quantity, which may change in time. The amount is an *extensive* state: the energy in a wall, the mass of water in a tank, the charge on a capacitor. Doubling the place doubles the state.

The second question identifies the *paths*. Along a path the quantity flows at some rate: watts, kilograms per second, amperes. What drives the flow is a difference in an *intensive* quantity between the two ends of the path: temperature, pressure, voltage. Doubling the place does not double the intensive quantity; it is a property of the state at a point, not of the amount.

The third question identifies the *boundary*. Some paths lead to places the model does not describe. The quantity that crosses those paths enters or leaves the system. The places on the far side are held at imposed conditions, either an imposed intensive value (the room temperature, the grid voltage) or an imposed flow (a heat load, a scheduled injection).

Table 1.1: Four conservation domains and their three quantities.

Domain	Extensive state	Flow	Intensive driver
Mass (water network)	mass M [kg]	mass rate $\dot{m}$ [kg/s]	head h [m]
Energy (thermal)	internal energy E [J]	heat rate Q [W]	temperature T [K]
Charge	charge q [C]	current I [A]	voltage V [V]
Power (DC network)	—	line flow F [W]	angle θ [rad]

Table 1.1 lists the four conservation domains that recur in this book with their extensive state, their flow, and the intensive quantity that drives it. The rows are physically different; the columns are the same three questions. That parallel is the foundation of everything that follows. A method that works on the columns works on every row.

Two rows are worth a comment. The first is the one this book works in. For a pressurized water network the driver is written as *head*, metres of water column, rather than as pressure in pascals: the two differ by the constant ρg, and head is what a gauge on a main is calibrated in and what a pump curve is quoted in. Every statement in this book about a potential applies to it unchanged.

The last row is there to make the generality visible. In the DC approximation of a power network the conserved quantity is real power, the flow on a line is the power it carries, and the driver is the voltage angle difference between its ends. There is no extensive state because the approximation is steady: buses do not store power. The row still fits the three questions; the first answer is simply "nowhere". Nothing in this book is specialized to water or to heat, and the occasional appearance of another domain in a table is the reminder.

1.2 Nodes conserve

The three questions define a graph. The places are nodes; the paths are edges. Write $\mathcal{N}$ for the set of nodes, $\mathcal{E}$ for the set of edges, and $\mathcal{D}$ for the set of conservation domains the model tracks. A model with heat and water flowing through the same pipework has $\mathcal{D} = \{\text{energy}, \text{mass}\}$, which is a district heating network and is where Part VI ends; a cold water distribution network has $\mathcal{D} = \{\text{mass}\}$ alone.

1.2.1 The balance law

At each node n and for each domain d in which n can accumulate, the rate of change of the extensive state equals what flows in, minus what flows out, plus what is generated, minus what is lost:

$$\boxed{\frac{\mathrm{d}X_{n,d}}{\mathrm{d}t} \;=\; \sum_{e \in \mathcal{E}(n)} A_{n,e}\, P_{d,e} \;+\; G_{n,d} \;-\; L_{n,d}} \tag{1.1}$$

Here $X_{n,d}$ is the extensive state of domain d at node n, $\mathcal{E}(n)$ is the set of edges incident on n, $P_{d,e}$ is the flow of d along edge e, $G_{n,d}$ is a source at the node and $L_{n,d}$ a loss. The coefficient $A_{n,e}$ is the sign that says whether edge e delivers to or draws from node n; it is defined in §1.2.3.

Equation (1.1) is the whole of physics that a node contributes. It is linear in the flows, always and exactly. It has no parameters. Whether the domain is energy or mass or charge, whether the node is a wall or a tank or a bus, the balance row looks the same. This is not a modeling

convenience. It is the reason the graph is the right object: the graph is the conservation skeleton, and every domain that the model tracks uses that same skeleton.

1.2.2 Node roles

Not every node accumulates. Three roles cover the cases.

Balanced. The node has an extensive state that can change. Equation (1.1) holds with its left side live. A wall, a water tank, a battery.

Zero-storage. The node is a junction or a bus. It has no extensive state, so equation (1.1) holds with $\mathrm{d}X_{n,d}/\mathrm{d}t \equiv 0$: what comes in goes out. A pipe tee, an electrical bus, a mixing point.

Reservoir. The node is on the far side of the boundary. Its intensive state is imposed, not solved. It contributes no balance row, because the model does not claim to know its balance; it appears only as the far end of the edges that cross the boundary. The atmosphere, the utility grid, a feed line.

The first two roles are the inside of the system. The third is the outside. Which nodes are reservoirs is the modeler's first and most consequential decision, and §1.4 returns to it.

A node also declares which domains it is balanced in. A pipe carrying water may need a mass balance at each tee but no energy balance if the pipe is adiabatic; a bus needs a power balance and nothing else. The balance rows exist only where the modeler asks for them. The roles are declared per domain as well: a return manifold whose pressure is held by a pump is a reservoir for mass and yet a balanced node for energy, because the temperature of the water leaving it is the model's to determine. Example 1.3 has such a node.

1.2.3 Incidence

Give every edge a direction, from a tail node to a head node. The direction is a bookkeeping convention, not a physical claim: a flow that runs against the edge's direction is simply negative. The *incidence matrix* $\mathbf{A}$ has one row per non-reservoir node and one column per edge, with entries

$$A_{n,e} = \begin{cases} +1 & \text{edge } e \text{ enters } n \text{ (n is its head)}, \\ -1 & \text{edge } e \text{ leaves } n \text{ (n is its tail)}, \\ 0 & \text{otherwise.} \end{cases} \tag{1.2}$$

Every column of the full incidence matrix (with reservoir rows included) has one $+1$ and one -1, so the columns sum to zero. Dropping the reservoir rows breaks that property for the columns of boundary edges, and that is exactly right: the flow on a boundary edge is not balanced by any row of the model. It is what the system exchanges.

With $\mathbf{A}$ in hand, the sum in equation (1.1) is one row of a matrix product. Stack the flows of domain d into a vector $\mathbf{P}_d$ and the balances into a vector; the balance law for all nodes at once is $\mathrm{d}\mathbf{X}_d/\mathrm{d}t = \mathbf{A}\,\mathbf{P}_d + \mathbf{G}_d - \mathbf{L}_d$.

> **Principle 1.1.** *The incidence matrix is shared by every domain. Energy and mass in the same pipework do not have two graphs. They have one graph, and each domain contributes balance rows on the nodes where it is tracked, using the same* $\mathbf{A}$.

An edge that carries one domain but not another, such as a rotating shaft that transmits energy but no mass, or a conduction path through a solid, simply has no flow variable for the absent domain. Its column contributes zero to that domain's balance rows.

Table 1.2: Four closure patterns. Each is written as a residual that vanishes when the relation holds.

Pattern	Residual	Typical use
Carrier flow	$P_e - q_e\,\varepsilon_e$	a quantity carried by another: $P = \dot{m}\,h$ (enthalpy carried by mass), $P = I\,V$ (power carried by current)
Gradient flow	$P_e - g(\phi_i, \phi_j; \theta)$	driven by a difference in intensive state: $Q = U\,A\,(T_h - T_c)$, $F = B\,(\theta_i - \theta_j)$
Property	$y - f(x_1, \ldots, x_k)$	a property table or a device specification: $h = h(p, T)$, $Q = \mathrm{COP} \cdot W$
Piecewise	$y - f_b(x)$	a device that changes regime: a valve that saturates, a limiter, a breaker

1.3 Edges close

The balance law says that the flows at a node sum to the accumulation. It does not say what any flow is. That is the job of the edge.

1.3.1 Closure relations

Along each edge, for each domain it carries, there is a flow variable and a relation that ties that flow to the states at the edge's ends and to some parameters. The relation is called a *closure* because it closes the system of equations: the balance rows alone have more unknowns than rows, and the closures supply the missing rows.

Definition 1.1 (Closure)**.** A closure is a local relation $r(\mathbf{z}_{\mathrm{loc}}; \theta) = 0$ among a small set of variables $\mathbf{z}_{\mathrm{loc}}$, with parameters θ, written in *residual form*: a function that is zero exactly when the relation holds. The closure declares which variables it reads.

Four patterns cover the great majority of physical closures (Table 1.2).

The gradient pattern is the one the three questions anticipated: a flow driven by an intensive difference. Conduction, convection, DC current, DC power flow and laminar pipe flow all take this form, with different functions g and different parameters θ: a conductance, a heat transfer coefficient times an area, a susceptance. The carrier pattern appears whenever one domain rides on another, as enthalpy rides on mass flow. The property pattern is a catch-all for relations among variables at a single place. The piecewise pattern is the first hint of something the deterministic model handles awkwardly and the probabilistic model handles naturally; it returns in Chapter 7. Appendix D catalogues the closures of the domains this book touches. For each it gives the residual and its pattern, the unknowns and parameters, the Jacobian, where the relation kinks, its shape, and the parameter that usually carries the prior.

1.3.2 A closure is a relation, not an assignment

Write the conduction closure as most textbooks do:

$$Q = U\,A\,(T_h - T_c). \tag{1.3}$$

The notation suggests a computation: given the two temperatures, compute the heat rate. But nothing in the physics says which quantities are given. If the heat rate and the hot temperature are known, the same relation determines the cold temperature:

$$T_c = T_h - Q/(UA). \tag{1.4}$$

If all three are measured, the relation determines the conductance UA. The physics asserts that four quantities satisfy one constraint. It does not assert a direction.

The residual form makes this explicit:

$$r(Q, T_h, T_c; UA) \;=\; Q - UA\,(T_h - T_c) \;=\; 0. \tag{1.5}$$

Equation (1.5) has no left-hand side to compute. It is a surface in the space of its variables, and every point on the surface is a physically admissible state of the edge.

> **Principle 1.2.** *A closure relates its variables. It does not compute one of them from the others. Which variable is unknown is a property of the question being asked, not of the physics.*

This principle looks like a remark about notation. It is not. It decides, in Chapter 2, which kind of graphical model a physical system should be written in, and it is the reason the arrows of a Bayesian network are not the natural representation of a closure. Hold on to it. The principle itself is old: it is the acausal view of physical laws that bond-graph modeling and equation-based modeling languages are built on [63, 69], and §1.8 says more.

1.3.3 Parameters

The parameters θ of a closure, a conductance, a susceptance, an efficiency, are ordinarily numbers the modeler supplies. Nothing forces that. A parameter the modeler does not know can be left as an unknown, added to the vector of variables, and determined by the equations if enough other things are known. A conductance inferred from a measured heat rate and two measured temperatures is the simplest case of this. The deterministic model needs one extra equation for each parameter it frees; the probabilistic model of Part II needs only a prior.

1.4 The boundary

The boundary is the set of edges with a reservoir at one end. Everything on the reservoir's side is outside the model. This section says what the model knows about the outside, why that knowledge is exact in one specific sense, and why the boundary is the most important part of the system.

1.4.1 Reservoirs and imposed conditions

A reservoir contributes no balance row. It contributes an imposed condition on the boundary edge, of one of two kinds.

Imposed potential. The reservoir's intensive state is fixed: the room temperature, the grid voltage angle at the point of connection. The boundary edge's closure then determines the flow across it. This is the Dirichlet condition of boundary-value problems.

Table 1.3: Ports: the conjugate flow and potential pair at a boundary, by domain.

Domain	Flow across the boundary	Potential at the boundary
Hydraulic	mass rate $\dot{m}$	head h
Thermal	heat rate Q	temperature T
Electrical	current I	voltage V
DC power	line flow F	angle θ

Imposed flow. The flow across the boundary edge is fixed: a specified heat load, a scheduled generation injection. The balance row at the inside node then determines the inside potential. This is the Neumann condition.

Each boundary edge is thus a pair, a flow and its conjugate potential, one of which is imposed and the other of which the model determines. The pair is domain-independent (Table 1.3). Call the pair a *port*.

Definition 1.2 (Boundary-exchanging system)**.** A boundary-exchanging system is a conservation graph with at least one reservoir node. It exchanges conserved quantities with its environment through its ports. Its state is determined jointly by its internal balances and closures and by the conditions imposed at its ports.

Every model in this book is boundary-exchanging. A closed system, one with no reservoirs, is a limiting case that occurs in textbooks and almost nowhere else. The term is this book's own. Thermodynamics calls such a system *open*, and port-based modeling calls the places where it trades with its environment *ports* [95]. The definition adds only that the trade goes through reservoir nodes of a conservation graph, which is what lets Chapter 5 treat the ports as separators.

1.4.2 Exactness at a cut

Take any set S of non-reservoir nodes, and consider the edges that cross from S to its complement. Sum the steady balance rows over the nodes in S. Every edge with both ends in S appears twice in the sum, once with $+P_e$ at its head and once with $-P_e$ at its tail; it cancels. Every edge with one end in S appears once. What remains is

$$\sum_{e \text{ crossing } S} A_{n(e),e}\, P_{d,e} \;=\; -\sum_{n \in S} (G_{n,d} - L_{n,d}), \tag{1.6}$$

where $n(e)$ is the end of e inside S. In words: the net flow of domain d across the boundary of S equals the net source inside S, whatever happens inside. Internal detail has no effect on the cut flow.

Proposition 1.1 (Exactness at a cut)**.** *In steady state the sum of the flows crossing the boundary of any node set S equals the net source inside S. The sum depends on S only through its sources and losses, not through its internal structure.*

This is a two-line consequence of conservation, and it is the reason a boundary can be trusted. Whoever sits outside S and sees only the port flows sees a quantity that is exact, not an approximation of the inside. A room controller that reads the total heat delivered by the racks it contains reads a number that is identically the cooling load, not an estimate of it. Chapter 17 builds an entire theory of hierarchical modeling on Proposition 1.1.

Note what the proposition does not say. It says nothing about the intensive states. The temperature at the boundary of S is not a sum of anything inside S, and no conservation law licenses aggregating it. Extensive flows aggregate by summation across a cut because a conservation law says so. Intensive states must be solved for.

1.4.3 Why the boundary matters most

Two reasons put the boundary at the center of this book.

The first is that it is unavoidable. The modeler who refuses to draw a boundary must model the universe. Every practical model draws one, and what it draws determines what the model can and cannot say. A building model with the weather as a reservoir cannot say anything about the weather; it can only say what the building does given the weather.

The second reason is that the boundary is where knowledge is thinnest. Inside the system the modeler has chosen every closure and knows every parameter, or at least knows which ones are uncertain. At the boundary the modeler has an imposed condition that summarizes an entire unmodeled world into one number per port. That number is a forecast, a schedule, a nominal value, or a measurement taken somewhere else. It is where the model's ignorance is concentrated.

When uncertainty enters in Part II it therefore enters at the boundary first. And when Part III asks how a large system can be solved as a collection of small ones, the answer is that each piece is a boundary-exchanging system in its own right, that its ports carry all that its neighbors need to know about it, and that Proposition 1.1 is what makes the ports sufficient. The machinery of separators and messages in the later chapters is the boundary of this section made probabilistic.

> **Principle 1.3.** *A system is defined by what crosses its boundary. Everything inside is the modeler's to describe; everything outside is summarized, exactly for the flows and by assumption for the potentials, at the ports.*

1.5 The residual system

Collect the pieces. The unknowns of the model are gathered into a single vector $\mathbf{z} \in \mathbb{R}^n$: the intensive state at each inside node, the flow on each edge channel, the extensive state at each balanced node, and any parameters the modeler has freed. Conditions imposed at reservoirs either remove a variable from $\mathbf{z}$ by substituting its value, or add an explicit row $z_k - \bar{z}_k = 0$; either way they are known.

The equations are gathered into a single vector-valued residual $\mathbf{F}(\mathbf{z}) \in \mathbb{R}^m$, whose rows come in a fixed order:

1. balance rows, one per (inside node, tracked domain) pair, equation (1.1) with the accumulation term moved to the right;
2. edge-channel closures, one per edge per domain it carries;
3. storage relations, one per extensive state, tying it to the intensive state at the same node ($E = C\,T$ for a thermal mass, for instance);
4. freestanding closures: properties, specifications, couplings not tied to one edge;
5. explicit boundary-condition rows.

The model is the system

$$\boxed{\mathbf{F}(\mathbf{z}) = \mathbf{0}} \tag{1.7}$$

in steady state, and $\mathbf{F}(\mathbf{z}, \dot{\mathbf{z}}) = \mathbf{0}$ with the accumulation terms live in the dynamic case, which is deferred to Chapter 16.

1.5.1 Well-posedness

Before anything is solved, the system can be checked for shape. It should be square, $m = n$: as many equations as unknowns. Its Jacobian $\mathbf{J} = \partial\mathbf{F}/\partial\mathbf{z}$ should have full structural rank, which means that some assignment of equations to unknowns covers every unknown; this is a property of which variables each row reads, not of their values, and it can be checked on the graph alone. The graph should be connected. Every flow variable should be constrained by some closure, since a balance row alone cannot determine two flows. No variable should be absent from every row.

Each of these checks fails in a way that points at a node, an edge or a variable. A model that fails one is not wrong in a numerical sense. It is underspecified or overspecified, and the failure says where. Making the checks structural, and making them precede any numerical work, is the difference between a modeling error that is reported as "node 7 has no closure on its outlet flow" and one that is reported as a singular matrix somewhere inside a solver.

The first check, squareness, can be done by counting before a single row is written.

Proposition 1.2 (Squareness by counting)**.** *Let a model have N inside nodes each balanced in every domain it touches, E edge channels (one per edge per domain the edge carries), S storage states, Θ freed parameters, C freestanding closures and R explicit boundary-condition rows, and let every inside node carry one potential per domain in which it is balanced. Then the number of unknowns exceeds the number of rows by exactly $\Theta - C - R$. The model is square if and only if every freed parameter is paid for by one freestanding closure or one explicit boundary condition.*

Proof. Count unknowns: one potential per (node, domain) pair, N_d in all; one flow per edge channel, E; one extensive state per storage, S; and Θ parameters. Count rows: one balance per (node, domain) pair, N_d; one closure per edge channel, E; one storage relation per storage, S; then C and R. The first three pairs cancel term by term, leaving $\Theta - (C + R)$. □

The proposition explains a pattern the examples below repeat. A model built from balanced nodes and closed edges is square by construction; it becomes non-square only when the modeler frees a parameter without adding a relation that pins it, or adds a relation without freeing anything. The first is the situation of a model with an unknown conductance and no measurement; the second, of a model with a measurement and nothing for it to determine. Both are exactly the situations that Chapter 3 will handle without counting, because a prior is a row that costs nothing and a measurement is a row that need not be paid for; but in the deterministic model the count must balance, and the proposition says how.

1.5.2 Sparsity

Each balance row has one nonzero entry per incident edge channel and one for its own storage state. Each closure row has one nonzero per variable it declares. The number of nonzeros in $\mathbf{J}$ is therefore of order $|\mathcal{E}|$ plus the total number of closure inputs: linear in the size of the model, not quadratic in the number of unknowns. A network with ten thousand nodes has a Jacobian with tens of thousands of nonzeros, not a hundred million.

The balance rows are exactly linear in the flows, so their entries in the flow columns are the constants $-A_{n,e}$ and never change. All of the nonlinearity, and all of the domain-specific

derivative work, lives in the closure rows. The pattern of nonzeros is known before any number is computed, from the closures' declared inputs alone.

Both facts, linear sparsity and a fixed pattern, are what make everything downstream affordable: solving equation (1.7) by Newton's method, eliminating the inside of a subsystem to expose its ports, and, in Part II, forming the precision matrix of a Gaussian posterior, which turns out to be $\mathbf{J}^{\mathsf{T}}\mathbf{J}$ up to weighting and inherits this sparsity exactly.

1.6 Three worked examples

Three examples, two hydraulic and one carrying two conserved quantities at once, show every object of this chapter in a form small enough to solve by hand or nearly so. The second is the running example of the book; it acquires a loop here, uncertainty in Chapter 3, and a closed pipe in Chapter 7. The third carries two domains on one graph, which is what a heating network does and what Part VI ends on. Every number is from `ch01_examples.py` in the book's `scripts` directory.

Example 1.1 (A pumping station lifting into a trunk main)**.** A borehole pumping station delivers a known flow G into the discharge manifold of its rising main (node 1). The water runs from there through a section of rising main, of conductance k, to an intermediate junction (node 2), and from that junction through a second section, of conductance k_2, into a trunk main (node a) that the station feeds. The second section carries a throttling valve, which is the one control the station has over how hard it has to pump, and k_2 is what the valve's position sets; Chapter 13 intervenes on it. The trunk main's head h_a is imposed: the trunk is the reservoir, and what the network beyond it does with the water is outside the model. Figure 1.1 draws the graph.

The numbers used throughout the book are $G = 100$ kg/s, $k = 5$ and $k_2 = 2$ kg/s per metre of head, with $h_a = 20$ m, which put the junction at 70 m and the pump's discharge at 90 m. The station therefore lifts seventy metres against the trunk main's twenty, and it is the second section — the one with the valve in it — that costs fifty of those seventy. That is the whole of the example's operational content, and the later chapters ask what can be known about it from gauges rather than from a flow meter

Unknowns. Two flows, m_{12} and m_{2a}, and two potentials, h_1 and h_2. The trunk main's head is imposed and substituted out.

Incidence. Rows for nodes 1 and 2, columns for edges e_{12} and e_{2a}, directions as drawn:

$$\mathbf{A} = \begin{pmatrix} -1 & 0 \\ +1 & -1 \end{pmatrix}. \tag{1.8}$$

The reservoir row $(0, +1)$ is dropped; the second column no longer sums to zero, which is the signature of a boundary edge.

Residuals. Two rows come from the balance law. At each inside node, equation (1.1) in steady state is $0 = \sum_e A_{n,e} Q_e + G_n$, written as the residual $r_n = -\sum_e A_{n,e} Q_e - G_n$, which vanishes when the node balances:

$$r_1 = -(-m_{12}) - G = m_{12} - G \qquad \text{(balance at node 1)}, \tag{1.9}$$

$$r_2 = -(m_{12} - m_{2a}) = m_{2a} - m_{12} \qquad \text{(balance at node 2)}. \tag{1.10}$$

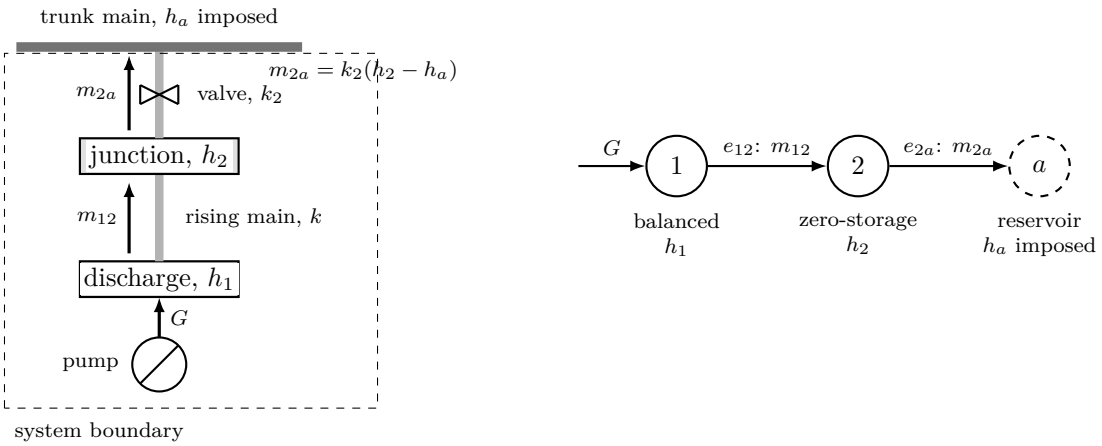


Figure 1.1: The pumping station of Example 1.1. Left: the physical arrangement. A pump delivers G into the discharge manifold at h_1; the water runs up a section of rising main, of conductance k, to a junction at h_2; a second section, throttled by a valve of conductance k_2, carries it into the trunk main at h_a. The dashed box is the system boundary; the network beyond the trunk is not modeled. Right: the same arrangement as a conservation graph. Node 1 is balanced (in steady state its accumulation is zero), node 2 is a junction, and node a, dashed, is the trunk reservoir on the far side of the boundary. Two edges, two closures; the source G is a known injection at node 1.

Two more come from the closures, one gradient closure on each edge:

$$r_3 = m_{12} - k\,(h_1 - h_2) \qquad \text{(closure on } e_{12}\text{)}, \tag{1.11}$$

$$r_4 = m_{2a} - k_2\,(h_2 - h_a) \qquad \text{(closure on } e_{2a}\text{)}. \tag{1.12}$$

Four rows, four unknowns, and every flow has a closure.

Vector form. Stack the unknowns, flows first, and the residuals in the order of the rows above:

$$\mathbf{z} = \begin{pmatrix} m_{12} \\ m_{2a} \\ h_1 \\ h_2 \end{pmatrix} = \begin{pmatrix} \mathbf{m} \\ \mathbf{h} \end{pmatrix}, \qquad \mathbf{F}(\mathbf{z}) = \begin{pmatrix} r_1 \\ r_2 \\ r_3 \\ r_4 \end{pmatrix} = \begin{pmatrix} m_{12} - G \\ m_{2a} - m_{12} \\ m_{12} - k\,(h_1 - h_2) \\ m_{2a} - k_2\,(h_2 - h_a) \end{pmatrix}.$$

The model is $\mathbf{F}(\mathbf{z}) = \mathbf{0}$, equation (1.7) for this example. Its rows come in two blocks, and the incidence matrix writes each block in one line. Collect the data of the model:

$$\mathbf{s} = \begin{pmatrix} G \\ 0 \end{pmatrix}, \qquad \mathbf{K} = \begin{pmatrix} k & 0 \\ 0 & k_2 \end{pmatrix}, \qquad \mathbf{A}_b = \begin{pmatrix} 0 & +1 \end{pmatrix}.$$

The vector $\mathbf{s}$ holds the sources at the inside nodes, the diagonal matrix $\mathbf{K}$ holds one conductance per edge, and $\mathbf{A}_b$ is the dropped reservoir row, through which the imposed head h_a enters. The balance rows r_1, r_2 are $-\mathbf{A}\mathbf{m} - \mathbf{s}$. For the closure rows, look at the vector $\mathbf{A}^\mathsf{T}\mathbf{h} + \mathbf{A}_b^\mathsf{T} h_a$. Its first entry is $-h_1 + h_2$ and its second is $-h_2 + h_a$: it is the head rise along each edge, head minus tail, so its negative is the drop that drives the flow. The closure rows r_3, r_4 are therefore

$\mathbf{m} + \mathbf{K}(\mathbf{A}^\mathsf{T}\mathbf{h} + \mathbf{A}_b^\mathsf{T} h_a)$, and

$$\mathbf{F}(\mathbf{z}) = \begin{pmatrix} -\mathbf{A}\mathbf{m} - \mathbf{s} \\ \mathbf{m} + \mathbf{K}(\mathbf{A}^\mathsf{T}\mathbf{h} + \mathbf{A}_b^\mathsf{T} h_a) \end{pmatrix}. \tag{1.13}$$

Jacobian. Differentiate $\mathbf{F}$ with respect to $\mathbf{z}$, one block at a time. The balance block depends on the flows alone, through $-\mathbf{A}$, and not on the heads. The closure block depends on the flows through the identity and on the heads through $\mathbf{K}\mathbf{A}^\mathsf{T}$. So

$$\mathbf{J} = \frac{\partial \mathbf{F}}{\partial \mathbf{z}} = \begin{pmatrix} \partial \mathbf{F}_{\text{bal}}/\partial \mathbf{m} & \partial \mathbf{F}_{\text{bal}}/\partial \mathbf{h} \\ \partial \mathbf{F}_{\text{clo}}/\partial \mathbf{m} & \partial \mathbf{F}_{\text{clo}}/\partial \mathbf{h} \end{pmatrix} = \begin{pmatrix} -\mathbf{A} & \mathbf{0} \\ \mathbf{I} & \mathbf{K}\mathbf{A}^\mathsf{T} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 1 & 0 & -k & k \\ 0 & 1 & 0 & -k_2 \end{pmatrix}. \tag{1.14}$$

The balance rows are constant and the closure rows carry the parameters. The structural rank is full.

Solution. Every model in this book is solved the same way, by Newton's method on $\mathbf{F}(\mathbf{z}) = \mathbf{0}$. Each step replaces $\mathbf{z}$ by $\mathbf{z} - \mathbf{J}^{-1}\mathbf{F}(\mathbf{z})$, one linear solve with the Jacobian, and the steps repeat until the residual vanishes. Here every closure is linear, so $\mathbf{F}$ is exactly linear in $\mathbf{z}$, $\mathbf{J}$ is constant, and the first step lands on the solution from any starting point. With $G = 100$ kg/s, $k = 5$ and $k_2 = 2$ kg/s per metre, and $h_a = 20$ metres, one step from $\mathbf{z} = \mathbf{0}$ gives $\mathbf{m} = (100, 100)^\mathsf{T}$ and $\mathbf{h} = (90, 70)^\mathsf{T}$: both flows equal the station's delivery, the discharge stands at 90 metres and the junction at 70. The last fact holds for any numbers. Every column of the full incidence matrix sums to zero, so $\mathbf{1}^\mathsf{T}\mathbf{A} + \mathbf{A}_b = \mathbf{0}$; multiplying the balance rows by $\mathbf{1}^\mathsf{T}$ gives $\mathbf{A}_b\mathbf{m} = \mathbf{1}^\mathsf{T}\mathbf{s}$, which says that the flow into the reservoir, m_{2a}, is the total source G. That is Proposition 1.1 for $S = \{1, 2\}$.

The solution is also linear in the two inputs, the source and the trunk main. Collect them in $\mathbf{u} = (G, h_a)^\mathsf{T}$; then each head is a fixed combination of them,

$$h_1 = \mathbf{h}_1^\mathsf{T}\mathbf{u}, \quad \mathbf{h}_1 = \begin{pmatrix} 1/k_2 + 1/k \\ 1 \end{pmatrix} = \begin{pmatrix} 0.7 \\ 1 \end{pmatrix}; \qquad h_2 = \mathbf{h}_2^\mathsf{T}\mathbf{u}, \quad \mathbf{h}_2 = \begin{pmatrix} 1/k_2 \\ 1 \end{pmatrix} = \begin{pmatrix} 0.5 \\ 1 \end{pmatrix}.$$

Each head is the boundary head plus the resistances between it and the boundary times the flow, exactly as an engineer would write it down. These numbers and these two vectors return in Chapter 3, where the delivery and the trunk main become uncertain and a pressure logger on a head reads $\mathbf{h}_1^\mathsf{T}\mathbf{u}$ or $\mathbf{h}_2^\mathsf{T}\mathbf{u}$. What the residual form adds is that the same four rows answer the other questions without rearrangement: if h_1 is measured and G is unknown, free G, add it to $\mathbf{z}$, and the system is again square, as Proposition 1.2 says it must be when one parameter is freed and one row is added.

Example 1.2 (A three-junction water loop)**.** Three junctions of a distribution network are connected in a triangle by pipes of equal conductance k. Junctions 2 and 3 carry known nodal sources q_2 and q_3, negative for a demand. Junction 1 is the connection to a trunk main whose head is far better supported than anything inside, and it is taken as the datum, $h_1 = 0$; every other head is measured from it and is negative, a drawdown. Junction 1 is a reservoir: the model does not claim to know the balance of the network beyond it. Figure 1.2 draws the graph.

The closure, and one honest simplification. The head loss along a water pipe is not linear in its flow. The Hazen–Williams law that water engineers use [100] makes it go as $m^{1.852}$, and

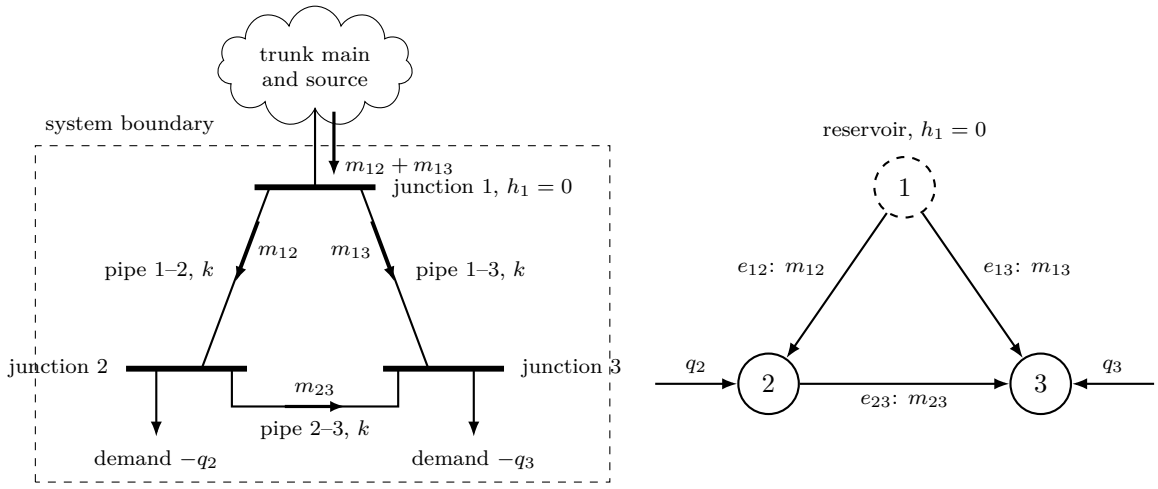


Figure 1.2: The water loop of Example 1.2. Left: the pipe schematic. Three junctions are joined by three pipes of equal conductance k; junctions 2 and 3 serve demands; junction 1 is connected to the trunk main, which lies outside the dashed system boundary and supplies whatever the demands need. Arrows mark the flow directions of the example. Right: the same network as a conservation graph. Junctions 2 and 3 are zero-storage; junction 1 is a reservoir at the datum head. The three pipes form a loop, the first loop in this book. Sources q_2, q_3 are imposed.

Chapter 8 and the studies of Part VI use it as it is. About an operating point, though, a pipe is linear to first order,

$$\Delta h \;=\; \Delta h_0 + \left.\frac{\mathrm{d}(\Delta h)}{\mathrm{d}m}\right|_{m_0} (m - m_0) \;\approx\; \frac{m}{k}, \tag{1.15}$$

with the offset absorbed into the datum and k the pipe's *conductance* at that point, in kilograms a second per metre of head. That is the model this example uses, and the model the running example of the book keeps using through Part III, because it is the one on which every number can be checked by hand. It is also a real method: solving a water network by repeatedly relinearizing its pipes about the current flows is the linear-theory method, and one iteration of it is exactly the system below. Nothing in Chapters 2 to 11 depends on the closure being linear; what depends on it is the reader's ability to check the arithmetic, which is worth more in those chapters than fidelity. Chapter 8 onward restores the exponent.

Unknowns. Three flows, m_{12}, m_{13} and m_{23}, and two heads, h_2 and h_3; h_1 is imposed and substituted out.

Incidence. Rows for junctions 2 and 3, columns for the three pipes in the directions drawn:

$$\mathbf{A} = \begin{pmatrix} +1 & 0 & -1 \\ 0 & +1 & +1 \end{pmatrix}. \tag{1.16}$$

Residuals. Two rows come from the balance law at junctions 2 and 3, which are zero-storage, so there is no accumulation:

$$r_2 = -(m_{12} - m_{23}) - q_2 \qquad \text{(balance at junction 2)}, \tag{1.17}$$

$$r_3 = -(m_{13} + m_{23}) - q_3 \qquad \text{(balance at junction 3)}. \tag{1.18}$$

Three more come from the closures, one gradient closure on each pipe:

$$r_{12} = m_{12} - k\,(h_1 - h_2) = m_{12} + k\,h_2 \qquad \text{(closure on } e_{12}\text{)}, \tag{1.19}$$
$$r_{13} = m_{13} + k\,h_3 \qquad \text{(closure on } e_{13}\text{)}, \tag{1.20}$$
$$r_{23} = m_{23} - k\,(h_2 - h_3) \qquad \text{(closure on } e_{23}\text{)}. \tag{1.21}$$

Five rows, five unknowns.

Vector form. As in Example 1.1, stack the unknowns flows first and the residuals in the order of the rows above:

$$\mathbf{z} = \begin{pmatrix} m_{12} \\ m_{13} \\ m_{23} \\ h_2 \\ h_3 \end{pmatrix} = \begin{pmatrix} \mathbf{m} \\ \mathbf{h} \end{pmatrix}, \qquad \mathbf{F}(\mathbf{z}) = \begin{pmatrix} r_2 \\ r_3 \\ r_{12} \\ r_{13} \\ r_{23} \end{pmatrix}.$$

With the sources $\mathbf{s} = (q_2, q_3)^{\mathsf{T}}$, one conductance per pipe, $\mathbf{K} = k\,\mathbf{I}$, and the reservoir row of junction 1, $\mathbf{A}_b = (-1, -1, 0)$, with its head h_1, the residual vector has the two blocks of equation (1.13), with the loop's heads in place of the chain's: the balance rows are $-\mathbf{A}\mathbf{m} - \mathbf{s}$ and the closure rows are $\mathbf{m} + \mathbf{K}(\mathbf{A}^{\mathsf{T}}\mathbf{h} + \mathbf{A}_b^{\mathsf{T}} h_1)$. Differentiating block by block, as there, gives the Jacobian

$$\mathbf{J} = \frac{\partial \mathbf{F}}{\partial \mathbf{z}} = \begin{pmatrix} -\mathbf{A} & \mathbf{0} \\ \mathbf{I} & k\,\mathbf{A}^{\mathsf{T}} \end{pmatrix}.$$

It is worth noticing what the two examples share. The pumping chain moves water along a path; this loop moves it around a ring; and the two blocks, the incidence of the balances and the conductance of the closures, are written with the same symbols because they are the same objects. That is the claim of this chapter, and it is now visible twice.

Solution. Newton's method on $\mathbf{F}(\mathbf{z}) = \mathbf{0}$ again. The closures are linear, so from $\mathbf{z} = \mathbf{0}$ it takes one step. With $q_2 = q_3 = -1$ (two equal demands, in kilograms a second), both heads are $-1/k$ metres and $\mathbf{m} = (1, 1, 0)^{\mathsf{T}}$. The pipe between the two demands carries nothing, by symmetry. The flow into the model across its boundary follows as in Example 1.1: here too $\mathbf{1}^{\mathsf{T}}\mathbf{A} + \mathbf{A}_b = \mathbf{0}$, so $\mathbf{A}_b\mathbf{m} = \mathbf{1}^{\mathsf{T}}\mathbf{s}$, which reads $-(m_{12} + m_{13}) = q_2 + q_3$. The boundary flow $m_{12} + m_{13} = 2$ is the total demand: Proposition 1.1 again, with $S = \{2, 3\}$. What the trunk main must deliver is precisely the port flow of the boundary-exchanging system, and in a domain that conserves mass this is not a convention but a fact a utility can meter.

Now break the symmetry: $q_2 = -1.5$, $q_3 = -0.5$. Then $\mathbf{m} = (1.167, 0.833, -0.333)^{\mathsf{T}}$. The third entry, m_{23}, is negative: a third of a kilogram a second flows from junction 3 to junction 2, against the drawn direction. The loop matters. Junction 2's extra demand is served partly by the direct pipe from junction 1 and partly around the other side of the triangle. There is no way to compute m_{23} from a chain of local steps; it depends on the whole loop at once. That dependence, harmless here, is the thing that makes inference on this graph interesting in Part III.

The nodal form. Because every closure here is linear, the flows can also be eliminated by hand, and that is how a water engineer usually writes such a network. The closure rows give $\mathbf{m} = -\mathbf{K}(\mathbf{A}^{\mathsf{T}}\mathbf{h} + \mathbf{A}_b^{\mathsf{T}} h_1)$. Substituting this into the balance rows, with $h_1 = 0$, leaves two rows in the two heads, and their solution:

$$k \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \begin{pmatrix} h_2 \\ h_3 \end{pmatrix} = \begin{pmatrix} q_2 \\ q_3 \end{pmatrix}, \qquad \begin{pmatrix} h_2 \\ h_3 \end{pmatrix} = \frac{1}{3k} \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} q_2 \\ q_3 \end{pmatrix}, \tag{1.22}$$

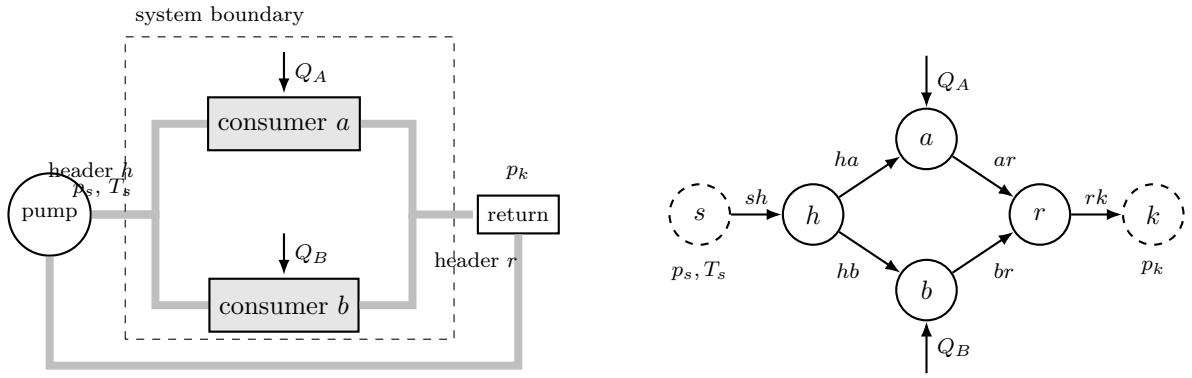


Figure 1.3: The district cooling loop of Example 1.3. Left: a pump supplies two consumers' substations in parallel through a header; each takes heat out of its building and into the water, and the water returns through a second header. Right: the same loop as a conservation graph, with the supply and suction sides of the pump as reservoirs. Every edge carries a mass channel and an energy channel, and the two domains use the same incidence matrix; the heat inputs are energy sources at the consumers' nodes.

after which $\mathbf{m} = -k\,\mathbf{A}^{\mathsf{T}}\mathbf{h}$. The matrix $k\,\mathbf{A}\mathbf{A}^{\mathsf{T}}$ on the left is the reduced conductance matrix of the network, and this elimination is the nodal form a hydraulic textbook starts from. The loop is visible in it. Without pipe e_{23} the incidence matrix would be its first two columns alone, $\mathbf{A}\mathbf{A}^{\mathsf{T}}$ would be the identity, and each head would follow from its own source; the off-diagonal -1 is the pipe that closes the loop, and it is why m_{23} depends on both sources at once. The elimination is a shortcut that linear closures allow, not a different model. The residual form contains it but does not require it, and once the closures stop being linear — which, in a real water network, they are not — the shortcut is gone: only Newton's method on the five-row system remains. That is why Chapter 8 and Part VI, which use the true exponent, never write a nodal form at all.

Example 1.3 (A district cooling loop carrying mass and energy)**.** A pump delivers chilled water at an imposed pressure p_s and temperature T_s to a supply header (node h). From the header two branches run through the substations of consumers a and b, which take heat Q_A and Q_B out of the buildings behind them and put it into the water, to a return header (node r), from which the water returns to the pump's suction at an imposed pressure p_k. Figure 1.3 draws the arrangement and its graph. It is the smallest instance of a network that carries two conserved quantities at once, and it is here to show that the constructions of this book do not care how many ledgers a graph has. Part VI does not use it: every study there is a single-ledger water network. A companion edition takes the two-ledger case up on district heating networks, where a four-consumer version of exactly this loop is worked against real data. Two conserved quantities move through the same pipes, mass and energy, and the point of the example is that they share one graph.

Unknowns. Four inside nodes, each with a pressure and a temperature, and six edges, each with a mass flow $\dot{m}_e$ and an energy flow P_e: twenty unknowns in all.

Incidence. Rows h, a, b, r, columns sh, ha, hb, ar, br, rk:

$$\mathbf{A} = \begin{pmatrix} +1 & -1 & -1 & 0 & 0 & 0 \\ 0 & +1 & 0 & -1 & 0 & 0 \\ 0 & 0 & +1 & 0 & -1 & 0 \\ 0 & 0 & 0 & +1 & +1 & -1 \end{pmatrix}, \tag{1.23}$$

used twice, once for the mass balances and once for the energy balances, which is Principle 1.1 in a matrix.

Residuals. Eight rows come from the balance law, with the one incidence matrix serving both domains: a mass balance and an energy balance at each inside node $n \in \{h, a, b, r\}$. The energy balance carries the heat input Q_n as a source, with $Q_a = Q_A$, $Q_b = Q_B$ and zero elsewhere:

$$r_n^{\dot m} = -\sum_e A_{n,e}\, \dot m_e \quad \text{(mass balance)}, \qquad r_n^{P} = -\sum_e A_{n,e}\, P_e - Q_n \quad \text{(energy balance)}.$$

Twelve more come from the closures, two on each edge e. The laminar closure is the gradient pattern of Table 1.2 in the mass domain. The carrier closure is the carrier pattern: the energy on the edge is its mass flow, times the specific heat c, times the temperature of the node it leaves.

$$r_e^{g} = \dot m_e - g_e\,(p_{\text{tail}} - p_{\text{head}}) \quad \text{(laminar closure)}, \qquad r_e^{c} = P_e - \dot m_e\, c\, T_{\text{tail}} \quad \text{(carrier closure)},$$

with the imposed values p_s, p_k and T_s wherever a tail or head is a reservoir. Twenty rows, twenty unknowns; Proposition 1.2 with $\Theta = C = R = 0$.

Vector form. As in Example 1.1, stack the unknowns flows first and the residuals balances first:

$$\mathbf{z} = \begin{pmatrix} \dot{\mathbf{m}} \\ \mathbf{P} \\ \mathbf{p} \\ \mathbf{T} \end{pmatrix},$$

with $\dot{\mathbf{m}}$ and $\mathbf{P}$ the mass and energy flows on the six edges and $\mathbf{p}$ and $\mathbf{T}$ the pressures and temperatures at h, a, b, r. The residual vector $\mathbf{F}(\mathbf{z})$ has four blocks, the two balances and then the two closures, and the model sets each to zero:

$$-\mathbf{A}\,\dot{\mathbf{m}} = \mathbf{0}, \qquad -\mathbf{A}\,\mathbf{P} - \mathbf{Q} = \mathbf{0}, \tag{1.24}$$

$$\dot{\mathbf{m}} + \mathbf{G}\,(\mathbf{A}^{\mathsf T}\mathbf{p} + \mathbf{A}_b^{\mathsf T}\mathbf{p}_b) = \mathbf{0}, \qquad \mathbf{P} - c\,\mathbf{M}\,(\mathbf{U}\mathbf{T} + \mathbf{u}_b\, T_s) = \mathbf{0}. \tag{1.25}$$

The pieces are these.

- $\mathbf{Q} = (0, Q_A, Q_B, 0)^{\mathsf T}$ holds the heat inputs at the nodes.
- $\mathbf{G} = \operatorname{diag}(g_{sh}, \ldots, g_{rk})$ holds the laminar conductances. The left closure is $\dot m_e = g_e\,(p_{\text{tail}} - p_{\text{head}})$ on every edge, the gradient pattern of Table 1.2 in the mass domain, written exactly as the chain's.
- $\mathbf{A}_b$ holds the reservoir rows of s and k, and $\mathbf{p}_b = (p_s, p_k)^{\mathsf T}$ their imposed pressures:

$$\mathbf{A}_b = \begin{pmatrix} -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & +1 \end{pmatrix}.$$

- The right closure is the carrier pattern: the energy on an edge is its mass flow, times the specific heat c, times the temperature of the node it leaves. $\mathbf{M} = \operatorname{diag}(\dot{\mathbf{m}})$ puts the mass

flows on a diagonal, and $\mathbf{U}$ picks each edge's tail temperature. Row e of $\mathbf{U}$ has a one in the column of e's tail when the tail is an inside node. The one edge that leaves a reservoir, sh, takes the imposed supply temperature T_s through the vector $\mathbf{u}_b$ instead:

$$\mathbf{U} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \qquad \mathbf{u}_b = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

Solution. Take $p_s = 300$ and $p_k = 100$ kilopascals, conductances of 0.05, 0.02, 0.03, 0.02, 0.03 and 0.05 kilograms per second per kilopascal on the six edges in the order above, $T_s = 20$ degrees, $Q_A = 20$ and $Q_B = 40$ kilowatts, and $c = 4180$ joules per kilogram-kelvin.

The model is solved by Newton's method on $\mathbf{F}(\mathbf{z}) = \mathbf{0}$, as the other two are, from a rough guess: every pressure 200 kilopascals, every temperature 20 degrees, every mass flow 1 kilogram per second and every energy flow 100 kilowatts. It converges in three steps. The largest residual falls from 1.0×10^5 to 9.0×10^4, then to 1.9×10^{-6} and 5.8×10^{-11}; the Jacobian below says why so few steps are needed.

The pressures are $\mathbf{p} = (250, 200, 200, 150)^{\mathsf{T}}$ kilopascals, and every edge drops the same 50. The mass flows are $\dot{\mathbf{m}} = (2.5, 1.0, 1.5, 1.0, 1.5, 2.5)^{\mathsf{T}}$ kilograms per second: 2.5 through the loop, split 1.0 through consumer a and 1.5 through consumer b. The temperatures are $\mathbf{T} = (20, 24.78, 26.38, 25.74)^{\mathsf{T}}$. The supply header stays at 20, since nothing heats it; the consumers sit $Q/(\dot{m}c)$ above the supply; and the return header is the flow-weighted mixture of the two consumers, as its energy balance requires. The energy flows are $\mathbf{P} = (209.0, 83.6, 125.4, 103.6, 165.4, 269.0)^{\mathsf{T}}$ kilowatts. With two reservoir rows the column identity of Example 1.1 reads $\mathbf{1}^{\mathsf{T}}\mathbf{A} + \mathbf{1}^{\mathsf{T}}\mathbf{A}_b = \mathbf{0}$, and multiplying the energy balances by $\mathbf{1}^{\mathsf{T}}$ gives $P_{rk} - P_{sh} = \mathbf{1}^{\mathsf{T}}\mathbf{Q}$. Energy enters across the boundary at 209 kilowatts and leaves at 269, and the difference is the 60 kilowatts of the two heat inputs. That is Proposition 1.1 in the energy domain. The mass balances, which have no sources, give $\dot{m}_{rk} = \dot{m}_{sh}$, the same proposition in the mass domain.

The carrier closures are bilinear, $\dot{m}_e T_{\text{tail}}$, so the model is the first nonlinear one in the book. Its Jacobian, with the unknowns in the order $(\dot{\mathbf{m}}, \mathbf{P}, \mathbf{p}, \mathbf{T})$ and the rows in the order of equations (1.24)–(1.25), is

$$\mathbf{J} = \begin{pmatrix} -\mathbf{A} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{A} & \mathbf{0} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} & \mathbf{G}\mathbf{A}^{\mathsf{T}} & \mathbf{0} \\ -c\,\mathbf{D} & \mathbf{I} & \mathbf{0} & -c\,\mathbf{M}\mathbf{U} \end{pmatrix}, \qquad \mathbf{D} = \operatorname{diag}(\mathbf{U}\mathbf{T} + \mathbf{u}_b T_s),$$

where $\mathbf{D}$ holds the tail temperatures. The nonlinearity sits in the last row alone, in the blocks $-c\,\mathbf{M}\mathbf{U}$ and $-c\,\mathbf{D}$, which carry the state. It is also one-directional. Put the mass unknowns and rows first: no mass row reads $\mathbf{T}$ or $\mathbf{P}$, so the Jacobian is block lower triangular, and its mass block is constant. The first Newton step therefore solves the mass problem exactly, whatever the guess: after it the pressures and mass flows are right to 10^{-8}, while the temperatures are still 8.6 degrees off. With $\dot{\mathbf{m}}$ right, the energy rows are linear in $\mathbf{T}$ and $\mathbf{P}$, and the second step solves them. Figure 1.4 draws the Jacobian's pattern: 53 nonzeros in a 20×20 matrix, the balance rows constant, the carrier rows reading a flow, a temperature and an energy flow each.

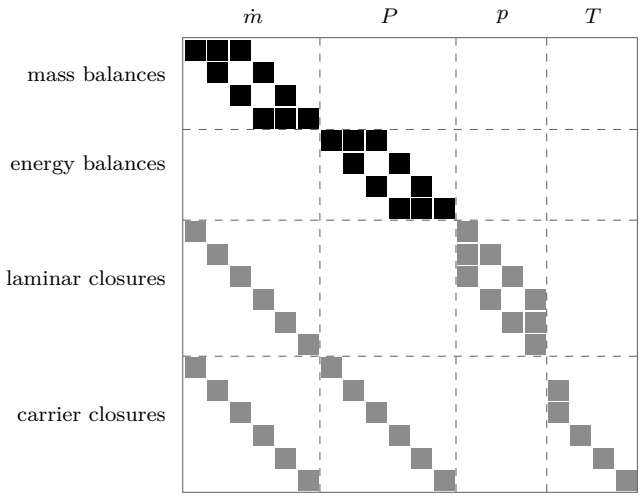


Figure 1.4: Nonzero pattern of the Jacobian of Example 1.3, rows in the fixed order of §1.5 and columns grouped by variable kind, flows first. Black entries are the constant ± 1 of the balance rows; grey entries carry parameters or the state. Thirteen percent of the matrix is nonzero, and the pattern was known before any number was.

Remark 1.1. All three examples have a Jacobian whose balance rows are constants and whose closure rows carry every parameter. All are square with full structural rank. All have a flow across the boundary that equals the net source inside. And in all, the unknowns could be reassigned, freeing a parameter and fixing a state, without changing a single row. The structure is the model. The numbers are an instance.

1.7 In practice

Draw the boundary first. Before any equation, decide which nodes are reservoirs. Everything the model will be unable to say follows from that decision, and so does everything it will be asked. Write the imposed condition at every port, potential or flow, and check that at least one port in every connected domain imposes a potential; a domain with only flows imposed has no level, as Exercise 1.1 shows.

One graph, several domains. Draw the nodes and edges once. Then, domain by domain, declare which nodes are balanced and which edges carry a channel. Do not draw a second graph for the second domain; the shared incidence matrix is what later lets a measurement in one domain inform the other.

Write every closure as a residual. Resist the assignment form. A closure written as $Q = UA(T_h - T_c)$ will be read as a computation, and a model that is a chain of computations cannot be run backwards, cannot absorb a measurement, and cannot free a parameter. A closure written as $Q - UA(T_h - T_c) = 0$ can.

Count before solving. Proposition 1.2 takes a minute and catches the two commonest errors: a freed parameter with nothing to pin it, and a measurement with nothing to determine. Then

check structural rank, which catches the subtler error of a subsystem that is square overall but has three rows on two unknowns in one place and two rows on three in another.

Check the cut. After any solve, sum the port flows of each domain and compare with the net source inside. The identity of Proposition 1.1 costs nothing to verify and fails loudly when a sign in the incidence matrix is wrong, which is the commonest error of all.

Keep the pattern. The Jacobian's nonzero pattern is fixed by the declared inputs of the closures. Record it. Everything that follows, the solver's ordering, the elimination of a subsystem onto its ports, the sparsity of the posterior precision, is a property of that pattern, and the pattern should be inspected once, as in Figure 1.4, before any of it is trusted.

1.8 What is missing

Every number in this chapter was known. The rising main's conductance was given; the valve's conductance was given; the trunk main's head and the nodal sources were imposed. Nothing was measured. Nothing failed.

List where a real model's ignorance actually sits.

- *Parameters.* The conductance of a real valve is a nominal value from its manufacturer's curve, correct to perhaps twenty percent. The conductance of a real pipe depends on a roughness that grows with the pipe's age and that nobody has measured since it was laid.
- *Boundary conditions.* The trunk main's head is a forecast. The demand at junction 2 is whatever the customers behind it happen to draw, and they have not been asked.
- *Model form.* The pipe from junction 2 to junction 3 is either open or shut — a closed valve, a burst under repair — and the closure is different in the two cases. A pressure-reducing valve is either controlling or wide open.
- *Measurements.* A real system has sensors. Some of its heads and flows are observed, with noise, and the deterministic model has no place to put an observation. It takes imposed values and returns solved values; it cannot take a noisy reading of a solved value and revise its other solved values in the light of it.

The last item is the decisive one. An observation of h_2 in Example 1.1 is information about k, about G, about h_a, and about h_1. The deterministic model can use it only by choosing which one unknown it will determine. The probabilistic model uses it for all of them at once, in proportion to how much each is uncertain.

Chapter 2 rewrites $\mathbf{F}(\mathbf{z}) = \mathbf{0}$ as a graph of relations, in which each row is a node in its own right that touches only the variables it reads. Chapter 3 attaches probability to that graph: a prior on each uncertain parameter and boundary condition, a likelihood on each measurement, and a factor for each row of $\mathbf{F}$. The joint distribution over every variable in the system is then a product,

$$p(\mathbf{z}) \;\propto\; \prod_{f} \varphi_f(\mathbf{z}_f), \tag{1.26}$$

in which each factor φ_f touches only a small set $\mathbf{z}_f \subset \mathbf{z}$. The set of factors is precisely the set of rows of this chapter, plus the priors and the sensors. Nothing about the structure changes. That is the claim the rest of the book makes good on.

Notes and further reading

The view of a physical system as conserved quantities at nodes, flows driven by potential differences along edges, and conjugate flow-and-potential pairs at ports, with the same structure across thermal, fluid, electrical and mechanical domains, is the bond-graph tradition begun by Paynter [69] and developed by Karnopp and Rosenberg [39]; what this chapter calls a port is their pair of effort and flow variables, and Table 1.3 is their table. Network thermodynamics [66] extended the same network structure to irreversible processes. The modern statement closest to §1.2 and §1.4, with an explicit incidence matrix, open graphs and compositional interfaces, is the port-Hamiltonian formulation on graphs of van der Schaft and Maschke [95]. The reader who knows any of these will recognize Chapter 1 as a restatement in their terms with the dynamics left for Chapter 16.

That a physical law is a relation and not an assignment (Principle 1.2) is the founding premise of acausal, equation-based modeling languages, of which Modelica [63] is the most widely used: its connectors carry potential variables that are equated and flow variables that are summed to zero at a connection, which is equation (1.1) generated automatically. The mathematical foundations of such languages, including the structural analysis that this chapter's well-posedness checks perform, are set out by Benveniste, Caillaud and Malandain [5]. Structural rank and the matching of equations to unknowns go back to Dulmage and Mendelsohn [20]; the same structural analysis applied to differential-algebraic systems is Pantelides' algorithm [68], and the numerical theory of such systems is in Brenan, Campbell and Petzold [12].

The sparsity argument of §1.5 and the claim that it is what makes everything downstream affordable are standard in the sparse-matrix literature; Davis [16] and Duff, Erisman and Reid [19] are the references, and the observation that power-network equations should be factored in an order chosen for sparsity is Tinney and Walker's [92]. The DC power flow of Example 1.2, its assumptions and its accuracy are reviewed by Stott, Jardim and Alsaç [87].

What is the book's own in this chapter is the emphasis: that the boundary is where uncertainty will enter, and that the exactness of cut flows (Proposition 1.1) will become the sufficiency of ports in Chapter 5.

Summary

- Three questions describe a conserved quantity in any system: where it accumulates, how it moves, what crosses the boundary. Their answers are nodes, edges and reservoirs.
- Nodes conserve. The balance law (1.1) is linear in the flows, has no parameters, and uses one incidence matrix shared by every domain.
- Edges close. A closure is a residual relation among a few variables; it does not assign a direction. Four patterns cover most of physics.
- The boundary is the set of edges to reservoirs. Each carries a port, a conjugate flow and potential pair. Cut flows are exact by conservation; cut potentials are not aggregable.
- The model is $\mathbf{F}(\mathbf{z}) = \mathbf{0}$: square, structurally full-rank, with sparsity linear in model size and a nonzero pattern fixed before any number is computed. It is square by construction, and stays square exactly when every freed parameter is paid for by a row.
- Two domains on one graph share the incidence matrix; the cooling loop's twenty rows carry mass and energy together, and the cut identity holds in each domain separately.
- Uncertainty sits in parameters, boundary conditions, model form and measurements. The

deterministic model has no place for a measurement. The probabilistic model of Part II will, on the same graph.

Exercises

Exercise 1.1. In Example 1.1, replace the imposed trunk-main head by an imposed flow $m_{2a} = \bar{m}$ at the boundary, and free the source G. Write $\mathbf{z}$ and $\mathbf{F}$. Is the system square? Compute its structural rank and identify the direction in which the Jacobian is singular. What single additional imposed condition restores a well-posed model, and why must it be a potential?

Exercise 1.2. Add an energy domain to Example 1.1 by giving the water a temperature, so that the flow into the trunk main carries heat by the carrier closure $Q_{2a} = m_{2a}\, c\, T_2$ with c the specific heat. Write the incidence matrix with both domains and confirm it is the same matrix. Which nodes need an energy balance?

Exercise 1.3. In Example 1.2, close pipe e_{23}. Solve for the heads and flows with $q_2 = -1.5$, $q_3 = -0.5$, and compare m_{12} with the looped case. Then close pipe e_{13} instead. Which closure changes the flow the trunk main must deliver at junction 1, and why does Proposition 1.1 say it cannot?

Exercise 1.4. Make junction 1 of Example 1.2 an inside zero-storage node with a known source q_1, and make there be no reservoir at all. Show that the resulting $\mathbf{F}$ is not structurally full-rank, identify the direction in which $\mathbf{J}$ is singular, and explain in words why a network with no imposed head cannot fix its own datum — a distribution system with every source modelled as a flow and none as a level.

Exercise 1.5. Prove Proposition 1.1 for the dynamic case: show that the net flow across the boundary of S equals the net source inside S minus the rate of change of the total extensive state inside S. Which quantity does a controller outside S then need to know in addition to the port flows?

Solutions to selected exercises

Exercise 1.1. The unknowns are now $\mathbf{z} = (m_{12}, m_{2a}, h_1, h_2, G, h_a)$, six of them, and the rows are the four of Example 1.1, with G and h_a moved into the unknowns, plus the boundary row $m_{2a} - \bar{m} = 0$: five rows. The system is not square, and Proposition 1.2 says why: two quantities were freed (G and h_a, so $\Theta = 2$) and one row was added ($R = 1$). Its Jacobian,

$$\mathbf{J} = \begin{pmatrix} 1 & 0 & 0 & 0 & -1 & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & -k & k & 0 & 0 \\ 0 & 1 & 0 & -k_2 & 0 & k_2 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix},$$

has rank five, and its null direction is $(0, 0, 1, 1, 0, 1)$: add the same constant to h_1, h_2 and h_a and every row is unchanged, because every head enters only through a difference. The flows are fully determined, $m_{12} = m_{2a} = G = \bar{m}$, and so are the head differences $h_1 - h_2 = \bar{m}/k$ and $h_2 - h_a = \bar{m}/k_2$; what is not determined is the datum. One imposed potential, any one of the

three temperatures, restores a square system of full rank; an imposed flow could not, because every flow is already fixed and a sixth flow row would either repeat one or contradict it. This is the thermal form of the floating-datum problem of the next exercise but one, and the general rule: a connected domain needs at least one port with an imposed potential, or its potentials float together.

Exercise 1.3. With $q_2 = -1.5$ and $q_3 = -0.5$ the looped network of Example 1.2 has $m_{12} = 1.167$, $m_{13} = 0.833$ and $m_{23} = -0.333$. Close e_{23}: each demand is served by its own pipe alone, $m_{12} = 1.5$, $m_{13} = 0.5$, and the heads are $h_2 = -1.5/k$, $h_3 = -0.5/k$ metres. Close e_{13} instead: junction 3 is served through junction 2, $m_{12} = 2.0$ and $m_{23} = 0.5$, with $h_2 = -2/k$ and $h_3 = -2.5/k$, the far junction now the lowest — which is what a utility means when it says a closed valve has left a street at the end of the line. In every case the flow the trunk main delivers, $m_{12} + m_{13}$, is 2.0, the total demand. Neither closure can change it, because Proposition 1.1 with $S = \{2, 3\}$ says the flow across the cut is the net source inside, and closing an internal pipe changes the internal structure and nothing else. What the closures change is how the two pipes from the reservoir share that flow, from 1.17 and 0.83 to 1.5 and 0.5 or to 2.0 and nothing, and that is the difference between the port flow, which conservation fixes, and the heads and internal flows, which the closures fix and any change inside can move. It is also, in one line, why a district meter tells a utility nothing about which of its pipes is doing the work.

Chapter 2

From equations to factor graphs

Chapter 1 ended with a system of equations, $\mathbf{F}(\mathbf{z}) = \mathbf{0}$, and a promise: that its structure is also the structure of every probabilistic question about the system. This chapter makes the structure visible. It draws the equations as a graph, compares the three graphical languages in which such a graph can be drawn, and argues that one of them, the factor graph, is the right one for physics. The argument turns on Principle 1.2 of the previous chapter: a closure relates its variables and does not compute one from the others.

There is still no probability in this chapter. Every factor is a relation that either holds or does not. Chapter 3 replaces "holds or does not" by "holds to within so much", and nothing drawn here will need to be redrawn.

2.1 The rows as a graph

Each row of $\mathbf{F}$ reads a few variables and ignores the rest. That pattern of reading is a graph.

Definition 2.1 (Factor graph of a residual system)**.** The factor graph of $\mathbf{F}(\mathbf{z}) = \mathbf{0}$ is a bipartite graph with one *variable node* for each component of $\mathbf{z}$, one *factor node* for each row of $\mathbf{F}$, and an edge between a factor node and a variable node exactly when that row reads that variable. The set of variables a factor reads is its *scope*.

The factor graph is nothing more than the sparsity pattern of the Jacobian $\mathbf{J} = \partial\mathbf{F}/\partial\mathbf{z}$ drawn as a graph: row k is adjacent to column j when J_{kj} is structurally nonzero. It was already known at compile time in §1.5, before any number was computed. What the drawing adds is that paths and loops become visible.

Figure 2.1 draws the pumping chain of Example 1.1. Two things are worth reading off it before going further.

First, the physical graph and the factor graph are different graphs of the same system. The physical graph had three nodes and two edges. The factor graph has four variable nodes and four factor nodes. A physical node became a balance factor. A physical edge became a flow variable together with a closure factor. The correspondence is systematic and is tabulated in §2.3; the point here is only that the two drawings answer different questions. The physical graph says where things are. The factor graph says which equations constrain which unknowns.

Second, the factor graph has a cycle, although the physical graph is a chain. Follow $h_2 \to c_{12} \to m_{12} \to b_2 \to m_{2a} \to c_{2a} \to h_2$. Each step is an edge of the figure. The cycle exists because there are two routes by which h_2 and m_{2a} are tied together: directly, through the

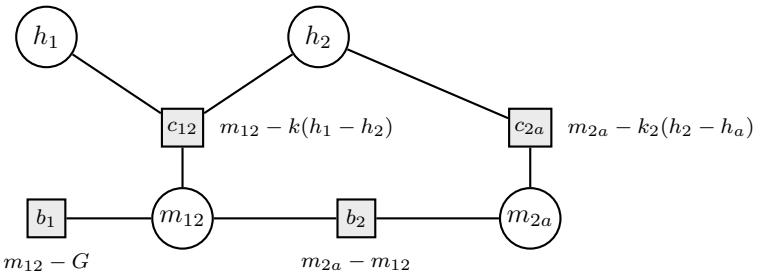


Figure 2.1: The pumping chain of Example 1.1 as a factor graph. Circles are the four unknowns; squares are the four rows of $\mathbf{F}$. The two balance rows b_1, b_2 read only flows. The two closures c_{12}, c_{2a} read a flow and the potentials at the ends of their edge. The known source G and the imposed room h_a are constants inside their factors, not nodes.

convection closure, and indirectly, through the conduction closure and the balance at node 2. Both hold at once. This is not a defect of the drawing. It is a fact about the equations, and it has consequences for inference that Part III will have to face: the factor graph of a physically tree-shaped system is not, in general, a tree.

2.2 Three languages

A factor graph is one of three standard ways to draw a function that splits into local pieces. All three were invented for probability distributions, and Chapter 3 will use them that way. But each is defined by the shape of a factorization, not by what the factors mean, so they can be compared now, on the deterministic system, where the comparison is sharpest.

Throughout this section, a *factorization* of a function Φ of many variables is a way of writing it as a product of functions, each of which reads only a subset of the variables. For the residual system the function is the indicator "every row is satisfied", which is the product over rows of the indicator "this row is satisfied". Each row is a local piece. The three languages differ in what they draw and what they leave out.

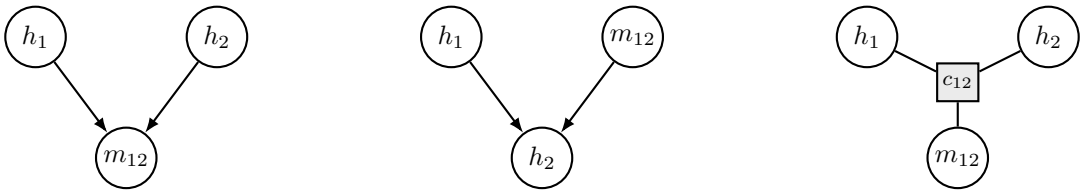
2.2.1 Directed graphs

A *directed graphical model*, or Bayesian network, draws each variable as a node and each factor as a set of arrows into one variable from the variables it depends on. The factorization it encodes is

$$\Phi(X_1, \dots, X_n) \;=\; \prod_i \phi_i(X_i \mid \mathrm{pa}(X_i)), \tag{2.1}$$

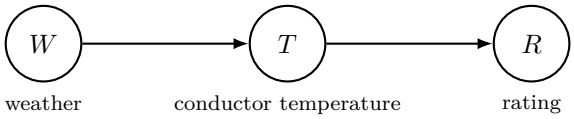
one factor per variable, each reading that variable and its *parents* $\mathrm{pa}(X_i)$, the sources of the arrows into it. The graph must have no directed cycle. When the factors are probabilities the form is $P(A)\,P(B \mid A)\,P(C \mid B)$ for the chain $A \to B \to C$, and the arrows read as "given".

The directed language is the natural one when the system has a generative story: something happens first, and something else depends on it. Weather drives the temperature of a conductor; the conductor's temperature sets its rating. The chain



(a) flow from heads (b) head from flow (c) the relation itself

Figure 2.2: Three drawings of one closure, $m_{12} - k(h_1 - h_2) = 0$. Panels (a) and (b) are directed and each commits to a computation; they encode different factorizations of the same relation and are both correct. Panel (c) is a factor graph and commits to nothing beyond the relation.



is a faithful picture. Each arrow points from a cause to an effect, the factorization $\phi(W)\,\phi(T \mid W)\,\phi(R \mid T)$ is the story told in order, and nobody would draw it the other way.

2.2.2 The directionality problem

Now try to draw a closure in the same language. The pipe closure of Example 1.2 is

$$m_{12} = k\,(h_1 - h_2). \tag{2.2}$$

The directed language needs one variable to be the child and the rest to be its parents. The textbook arrangement is $h_1 \to m_{12} \leftarrow h_2$, with k as a third parent if it is uncertain. But Principle 1.2 says the physics asserts no such thing. The closure is equally $h_2 = h_1 - m_{12}/k$, which is $h_1 \to h_2 \leftarrow m_{12}$, or $k = m_{12}/(h_1 - h_2)$, which is a third orientation — and the third is not a curiosity: it is what a utility does when it calibrates a pipe's roughness from a gauged head drop. Figure 2.2 draws the first two.

Each orientation is a correct picture of the same relation, and each encodes a different way of computing it. So the directed drawing is not a property of the physics. It is a property of the physics together with a choice of which quantities are known. That choice changes with the question, and the drawing changes with it.

This might be tolerable if some orientation always existed. It does not. Consider what an orientation of the whole system means: every row is assigned one output variable, no two rows share an output, and the arrows from inputs to outputs contain no directed cycle. An orientation with those properties is exactly an order in which the equations can be solved one at a time by substitution. Each row, taken in order, has all its inputs already computed and delivers its output.

For the pumping chain such an order exists. Read Figure 2.1: b_1 delivers $m_{12} = G$; b_2 then delivers $m_{2a} = m_{12}$; c_{2a} then delivers h_2; c_{12} finally delivers h_1. The directed drawing is $G \to m_{12} \to m_{2a} \to h_2 \to h_1$, and it is the order in which the example was solved by hand. But now change the boundary condition. Impose the heat rate $m_{2a} = \bar{Q}$ instead of the room temperature (Exercise 1.1). Then c_{2a} must deliver h_a, b_2 must deliver m_{12} from m_{2a}, and b_1 must deliver G. Every arrow reverses. The same physics, the same rows, and a different directed graph, because a different quantity was imposed at the boundary.

For the water loop no order exists at all. Try to build one. The balance at junction 2 reads m_{12} and m_{23} and must deliver one of them; the balance at junction 3 reads m_{13} and m_{23} and must deliver one of them. Suppose junction 2 delivers m_{12} and junction 3 delivers m_{13}. Then the closure c_{12} must deliver h_2 from m_{12}, the closure c_{13} must deliver h_3 from m_{13}, and c_{23} must deliver m_{23} from h_2 and h_3. But junction 2 needed m_{23} to deliver m_{12}, and m_{23} needs h_2, which needs m_{12}. That is a directed cycle. Every other assignment closes a cycle the same way; Exercise 2.2 asks for the full check. The loop in the network is an irreducible block of two equations in two unknowns, the 2×2 system of equation (1.22), and no sequence of single-equation substitutions solves it. A directed graph with one variable per node cannot represent it without first collapsing the block into one node that holds both heads, which is to say without giving up on drawing the structure at all.

The argument can be made exact, and the exact form is the one that modeling compilers use.

Proposition 2.1 (When an order of computation exists)**.** *A square residual system admits an order of single-row substitutions if and only if there is an assignment of one distinct output variable to each row such that the dependency graph, with an arrow from every input of a row to its output, has no directed cycle. Equivalently, the system's rows decompose into irreducible blocks, sets of rows that must be solved simultaneously, and an order exists exactly when every block has one row.*

Proof. If an order exists, take each row's output to be the variable it delivers; the outputs are distinct because each variable is delivered once, and the dependency graph has no cycle because every arrow points from a variable delivered earlier to one delivered later. Conversely, given such an assignment with an acyclic dependency graph, a topological order of that graph orders the outputs, and hence the rows, so that every row's inputs have been delivered before it is reached. For the second statement, fix any assignment, which is a perfect matching of rows to variables in the bipartite factor graph, and draw an arrow from row r to row s when s reads the variable matched to r. The strongly connected components of this graph are the blocks; they do not depend on which perfect matching was chosen, a fact due to Dulmage and Mendelsohn, and a block with more than one row is a directed cycle among rows that no matching removes. An order exists exactly when every component is a single row. □

The proposition turns the hand argument into a count, and the counts are in the chapter's script, `ch02_factor_graphs.py`, and in Table 2.1. The pumping chain has one assignment of outputs to rows, and its dependency graph is acyclic: the order found above is the only one. The water loop has three assignments, and every one has a directed cycle; its five rows form a single irreducible block. The cooling loop of Example 1.3, with twenty rows, has a block of ten, the four mass balances and six laminar closures, which is the mass network with its parallel branches; a block of three, the supply header's energy balance with the two carrier closures that leave it, which share the header temperature; and seven rows in blocks of one. Given the mass flows, the energy problem is nearly an order of substitutions, downstream node by downstream node, which is how a plant engineer computes it by hand; the mass problem is a loop and is not.

> **Principle 2.1.** *A directed graph encodes an order of computation. Physics supplies relations, not an order. Where an order exists it is fixed by the choice of boundary conditions and changes when they change; where the physics has a loop, no order exists.*

Table 2.1: The three examples of Chapter 1 as factor graphs: rows and variables, edges of the factor graph (nonzeros of $\mathbf{J}$), its cycle rank (edges minus nodes plus one), the sizes of the irreducible blocks of Proposition 2.1, and the edges of the Markov graph (off-diagonal nonzeros of $\mathbf{J}^\mathsf{T}\mathbf{J}$).

system	rows	variables	edges	cycle rank	blocks	Markov edges
pumping chain	4	4	8	1	$1, 1, 1, 1$	5
water loop	5	5	11	2	5	7
cooling loop	20	20	53	14	$10, 3$, seven of 1	46

2.2.3 Undirected graphs

An *undirected graphical model*, or Markov random field, draws each variable as a node and joins two variables by an edge whenever they appear together in some factor. The factorization it encodes is

$$\Phi(X_1, \ldots, X_n) \;=\; \prod_C \psi_C(X_C), \tag{2.3}$$

a product over *cliques* C, sets of mutually adjacent variables, of functions ψ_C that read only the variables in the clique. There are no arrows. A factor reads "these variables are jointly compatible", not "this one is computed from those". That is already the right reading for a closure, and it is why the undirected language is closer to physics than the directed one.

It still forgets something. The undirected graph of the pipe closure is a triangle on h_1, h_2, m_{12}: all three are pairwise adjacent because all three appear in one factor. But the same triangle is drawn for three separate pairwise relations, one on each pair, and for one three-way relation, and for a three-way relation together with a pairwise one. The graph records which variables interact and loses how many relations there are and which variables each one reads. For the water loop, where two closures both read h_2 and two balances both read m_{23}, the undirected drawing merges relations that the physics keeps distinct. Since the relations are the physics, this is the wrong thing to forget.

2.2.4 Factor graphs

A *factor graph* is Definition 2.1 with the factors allowed to be any local functions:

$$\Phi(X_1, \ldots, X_n) \;=\; \prod_f \varphi_f(X_f), \tag{2.4}$$

where X_f is the scope of factor f. Variables are circles, factors are squares, and a square is joined to exactly the circles in its scope. Nothing is oriented and nothing is merged. Panel (c) of Figure 2.2 is the pipe closure in this language: one square, three circles, and no claim about which is computed from which.

The factor graph keeps what the other two languages lose. It keeps the identity of each relation, which the undirected graph merges into cliques. It keeps the absence of direction, which the directed graph is forced to invent. And it is exactly the sparsity pattern of the Jacobian, so it costs nothing to construct: it is already there in the compiled model.

Principle 2.2. *A factor graph separates variables from the relations among them, and draws both. It is the language in which a residual system is its own picture.*

Table 2.2: What each physical object becomes in the factor graph, and what it will become when probability is attached.

Physical object	Residual system	Factor graph	Part II role
Balanced or zero-storage node	one balance row per tracked domain	a factor whose scope is the incident flows (and the node's storage state)	conservation factor
Edge channel	one flow variable and one closure row	a variable node and a factor whose scope is the flow, the end potentials, and the parameters	closure factor
Reservoir	a substituted constant, or an explicit boundary row	a constant inside the neighboring factors, or a unary factor on one variable	prior on a boundary condition
Freed parameter	a column of **J**	a variable node in the scope of every closure that reads it	prior on a parameter
Sensor	none	none yet	likelihood factor

Directed and undirected graphs are not wrong, and both will return. A directed graph is the right picture for a genuinely generative piece of a model, such as the weather chain above, and Chapter 3 attaches such pieces to the factor graph as priors. An undirected graph is the right picture for asking which variables are conditionally independent of which, and Chapter 5 uses it for exactly that. But the model is built, and inference is organized, on the factor graph.

2.3 The physical system as a factor graph

The correspondence noticed in Figure 2.1 is general. Table 2.2 states it: each object of the physical graph of Chapter 1 becomes a definite set of nodes in the factor graph. The last column anticipates Part II.

Two remarks on the table.

A reservoir can be drawn either way. Substituting its value makes it vanish into the factors that read it, as h_a vanished into c_{2a} in Figure 2.1. Keeping it as a variable with an explicit row $z_k - \bar{z}_k = 0$ makes it a variable node with one unary factor. The two are equivalent now, and the second becomes the natural one the moment $\bar{z}_k$ is uncertain, because the unary factor is then simply replaced by a prior. The factor graph is drawn once, and only the meaning of one square changes.

A sensor has no row in the deterministic system, which is the gap §1.8 identified. In the factor graph it will be a unary factor on the variable it observes. That is the whole of the change Chapter 3 makes to the structure: sensors are added as squares of degree one. The rest of the graph is untouched.

Figure 2.3 draws the water loop. It has the cycle traced in thick lines, a second cycle through h_3 on the right, and a long cycle that goes around the whole figure. The network's single physical loop has become several loops in the factor graph, and the pumping chain, which had no physical

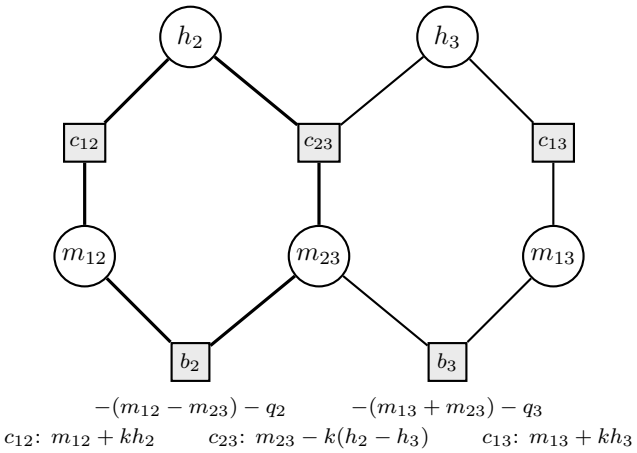


Figure 2.3: The water loop of Example 1.2 as a factor graph: five unknowns, five rows. The datum head $h_1 = 0$ and the injections q_2, q_3 are constants inside their factors. The thick edges trace one of the graph's cycles, $h_2 \to c_{12} \to m_{12} \to b_2 \to m_{23} \to c_{23} \to h_2$; there are others.

loop, had one. Counting physical loops does not predict the shape of the factor graph.

2.3.1 A junction is not a variable

The tempting simplification is that a physical node is a variable and a physical edge is a factor. It is a useful first intuition and it is wrong in both halves.

A junction of a cold water network has one unknown, its head, and one balance row; that is the only case where the simplification is nearly right, and even there the balance is a factor, not a variable. The same junction in a heating network has a head *and* a temperature, a mass source and a heat source, and two balance rows. A junction with a service reservoir on it has a stored mass and a storage row. A physical node becomes several variable nodes and one or more factor nodes.

A pipe in the same model has one flow and one closure. A pipe whose roughness is unknown adds a parameter variable, and a companion study on a laboratory rig infers thirty-six such parameters at once. A pipe in a network that carries heat as well as water needs a closure for each, with the temperature at its far end depending on its transit time and so on the flow the first closure sets. A pipe that may be shut adds a discrete availability variable, to which Chapter 7 is devoted. A physical edge becomes as many variable nodes and factor nodes as the physics of the edge requires.

Principle 2.3. *The physical topology guides the factorization; it does not dictate a one-to-one map. The closures decide which variables exist and which factors read them.*

2.3.2 Two domains on one graph

The cooling loop of Example 1.3 shows the map at its richest. Its factor graph has forty nodes, twenty variables and twenty factors, joined by fifty-three edges, one per nonzero of the Jacobian of Figure 1.4. Figure 2.4 draws the part of it that belongs to one edge of the physical graph,

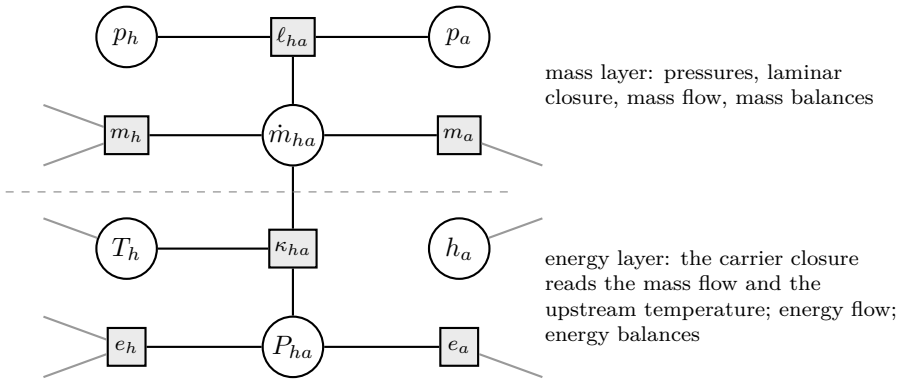


Figure 2.4: One branch of the cooling loop as a factor graph. Above the dashed line, the mass domain: the laminar closure ℓ_{ha} reads the two pressures and the mass flow, and the mass balances m_h, m_a read the flow. Below it, the energy domain: the carrier closure κ_{ha} reads the energy flow, the upstream temperature and the mass flow, and the energy balances read the energy flow. The mass flow $\dot{m}_{ha}$ is the only variable in both layers; it is where the domains couple, and it is why a temperature measurement can inform a pressure.

the branch from the supply header to consumer a, together with the balances at its ends. The pattern repeats for every edge.

The figure makes two points. The two domains are two layers of one graph, joined only through the mass flows, which the carrier closures read; a temperature measurement at consumer a constrains $\dot{m}_{ha}$ through κ_{ha}, and $\dot{m}_{ha}$ constrains the pressures through ℓ_{ha}. That path exists because the incidence matrix is shared. And the graph is loopy in both layers: the mass layer has the loop of the two parallel branches, which Proposition 2.1 found as a block of ten rows, and the energy layer, even though energy given mass is nearly an order of substitutions, has cycles through every node with two outgoing carriers. The cycle rank of the whole, edges minus nodes plus one, is fourteen.

2.4 The factor graph is not unique

Definition 2.1 draws one factor per row of $\mathbf{F}$. That is the finest factorization the model offers, and it is the canonical one for constructing the model. It is not the only one, and it is not always the one to compute with.

Take the water loop and eliminate the flows. Substitute each closure into the balances that read its flow, as was done to reach equation (1.22). The result is two rows in two unknowns,

$$k_2(h_2, h_3) = 2kh_2 - kh_3 - q_2, \qquad k_3(h_2, h_3) = -kh_2 + 2kh_3 - q_3, \tag{2.5}$$

and its factor graph has two variable nodes and two factor nodes, each factor reading both variables (Figure 2.5). The solution set is unchanged: the heads that satisfy $k_2 = k_3 = 0$ are exactly the heads of the five-row solution, and the flows are recovered from them by the closures. Three variables and three factors have disappeared from the drawing.

The same operation on the pumping chain eliminates m_{12} and m_{2a} and leaves two rows in h_1, h_2, each reading both. Both reduced graphs still have a cycle, a four-cycle through the two factors, because two factors read the same pair. Merge the two factors into one, which is always

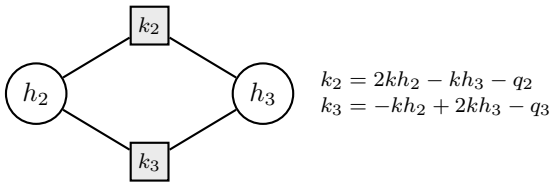


Figure 2.5: The water loop after eliminating its flows: the nodal form. Two variables, two factors, each reading both. It is a coarser factorization of the same solution set as Figure 2.3, and it still has a cycle, of length four, because two factors share the same scope.

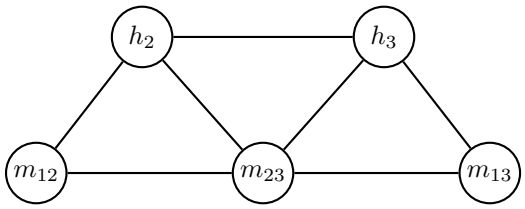
allowed since a product of two local functions with the same scope is one local function with that scope, and the chain's reduced graph is finally a tree: one factor on $\{h_1, h_2\}$. The triangle's reduced graph, merged, is a single factor on $\{h_2, h_3\}$, also a tree, and also trivial.

The cooling loop shows the same operation on a larger graph, and shows what it does to the Markov graph. Eliminate the six mass flows from the mass layer: substitute each laminar closure into the two mass balances that read its flow. What remains is four rows on the four pressures, each row reading its own node's pressure and its neighbors', with the supply and suction pressures as constants. The Markov graph of these four variables has four edges, h–a, h–b, a–r, b–r, and is the pipe network's own graph, a four-cycle. Before elimination the twenty variables had forty-six Markov adjacencies (Table 2.1); the pressures alone had none, since no row reads two pressures without a flow between them. Elimination created the four adjacencies by joining every pair of variables that shared a factor with an eliminated one. That is fill, the subject of Chapter 5, and here it is benign: it reproduces the physical network. The same substitution in the energy layer, given the flows, leaves four rows on the four temperatures with the network's edges directed downstream, which is why energy given mass is nearly an order of substitutions.

None of these is the wrong graph. Each is a factorization of the same function, and equation (2.4) is satisfied by all of them. They differ in how many variables they expose and how finely they split the relations, and those differences will govern the cost of inference. Chapter 5 shows that the cost of exact inference depends on a property of the factor graph called its treewidth, and treewidth is a property of the drawing, not of the solution set. Choosing the factorization is choosing the treewidth.

Principle 2.4. *The row-per-factor graph is canonical for building a model. Coarser factorizations, obtained by eliminating variables and merging factors, describe the same solution set with a different graph, and the graph is what inference pays for.*

There is a physical reading of elimination that is worth stating now, because Chapter 8 builds on it. Eliminating the flows of the loop produced the reduced conductance matrix, which is what a hydraulic textbook writes down first. Eliminating everything inside a subsystem except its ports produces a factor on the ports alone: the subsystem as its neighbors see it. That factor is the algebraic form of Proposition 1.1, and computing it is the Schur complement of Chapter 8. The boundary of Chapter 1 is a separator in the factor graph, and eliminating the interior is how a region is summarized to the rest of the system.



five variables, seven edges
h_2h_3, h_2m_{23}, h_3m_{23}: from c_{23}
h_2m_{12}: from c_{12}; h_3m_{13}: from c_{13}
$m_{12}m_{23}$: from b_2; $m_{13}m_{23}$: from b_3

Figure 2.6: The Markov graph of the water loop in row-per-factor form. Each edge is a pair of variables that some factor reads together; the closure c_{23}, with scope of size three, contributes a triangle. The graph is the sparsity pattern of $\mathbf{J}^\mathsf{T}\mathbf{J}$ for the Jacobian of Example 1.2, and Chapter 6 will show it is the sparsity pattern of the posterior precision.

2.5 Vocabulary

The rest of the book uses a small set of graph-theoretic terms about the factor graph. They are collected here.

Scope. The set of variables a factor reads. Written $\mathbf{z}_f$ for factor f.

Neighbors. The factors that read a variable, or the variables a factor reads. The two are the two ends of the bipartite edge set.

Degree. The number of neighbors. A prior or a sensor is a factor of degree one. A balance factor has degree equal to the number of incident edge channels plus one for a storage state.

Path, cycle, tree. As in any graph. A factor graph with no cycle is a tree; a factor graph with a cycle is *loopy*. All three examples of this chapter are loopy in row-per-factor form.

Cycle rank. The number of independent cycles, which for a connected graph is the number of edges minus the number of nodes plus one; zero for a tree. Table 2.1 lists it for the examples.

Markov graph. The undirected graph on the variables alone, in which two variables are adjacent when some factor reads both. It is the sparsity pattern of $\mathbf{J}^\mathsf{T}\mathbf{J}$, and it is the graph on which conditional independence is read in Chapter 5.

Separator. A set of variables whose removal disconnects the Markov graph. The ports of a subsystem are a separator between its interior and the rest of the system.

The Markov graph deserves more than a sentence, because the identification with $\mathbf{J}^\mathsf{T}\mathbf{J}$ is not a coincidence.

Proposition 2.2 (The Markov graph is the pattern of $\mathbf{J}^\mathsf{T}\mathbf{J}$)**.** *Two variables are adjacent in the Markov graph of a residual system if and only if the corresponding off-diagonal entry of* $\mathbf{J}^\mathsf{T}\mathbf{J}$ *is structurally nonzero.*

Proof. The entry $(\mathbf{J}^\mathsf{T}\mathbf{J})_{ij} = \sum_k J_{ki}J_{kj}$ is a sum over rows. A term is structurally nonzero exactly when row k reads both variable i and variable j, that is, when both are in the scope of factor k. The sum is structurally nonzero when at least one such row exists, which is the definition of adjacency in the Markov graph. (Numerical cancellation between rows could make the entry zero by accident; the structural statement ignores that, as sparsity analysis always does.) □

Figure 2.6 draws the Markov graph of the water loop: five variables, seven edges, one per pair that shares a factor, and the script confirms that the same seven pairs are the off-diagonal nonzeros of $\mathbf{J}^\mathsf{T}\mathbf{J}$. Chapter 6 shows that $\mathbf{J}^\mathsf{T}\mathbf{J}$, suitably weighted, is the precision matrix of the Gaussian posterior. The Markov graph of the physics is therefore the sparsity pattern of the

posterior's precision. That single fact is what makes inference on a physical system affordable at scale, and it was visible in this chapter, before any probability was introduced.

2.6 In practice

Build the row-per-factor graph and keep it. It is the Jacobian's pattern and costs nothing. Coarser graphs are derived from it when a solver needs them; the fine one is the record of what the model asserts.

Name every factor after its row. A factor called "balance at node 7" or "closure on line 12–13" can be pointed at when a check fails, a residual is large, or a measurement disagrees. A factor called φ_{41} cannot. The names of Table 2.2 are the vocabulary the rest of the book uses.

Keep reservoirs as unary factors. Substituting an imposed value into its neighbors is tidy and loses the place where the value will later become a prior. A variable with one square attached is the form that Chapter 3 needs, and it costs one node.

Do not let a junction be a variable. The commonest error of a reader arriving from graphical models is one circle per physical node and one square per physical edge. It is wrong for every model with more than one unknown per node, which is every heating network and every network whose junctions carry storage, and it hides the closures, which are where the physics and the parameters live.

Expect loops. A physically tree-shaped system has a loopy factor graph whenever a variable is tied to another by two routes, which is always, since a flow appears in a closure and in a balance at each end. Do not build inference on the assumption of a tree; build it on separators, which Chapter 5 supplies.

Run the block decomposition. Proposition 2.1's blocks say which parts of a model can be solved by substitution and which must be solved simultaneously, and modeling compilers compute them in linear time. A block that is unexpectedly large is usually a modeling error, a missing boundary condition or a closure written on the wrong variables, and the block names the rows involved.

2.7 What is missing

The factors of this chapter are indicators. Each is one where its row is satisfied and zero where it is not, and their product is one on the solution set and zero elsewhere. That is a legitimate factorization and the graph is a legitimate factor graph, but it is not yet a model of anything uncertain. The solution set of a square, well-posed system is a point.

Chapter 3 keeps every node and every edge of the factor graph and changes what the squares mean. A balance factor becomes "the balance holds to within a residual of this scale". A closure factor becomes the same for its relation. A reservoir's unary factor becomes a prior. A sensor, drawn for the first time, is a factor of degree one that says how far its reading may be from the variable it observes. The product of the squares is then a function that is large

where every relation nearly holds and small elsewhere, and after normalization it is a probability distribution over the whole state of the system. Every question in Part IV is a question about that distribution, and every one is answered on this graph.

Notes and further reading

Factor graphs in the form used here, a bipartite graph of variables and local functions with the product of the functions as the object of study, were defined by Kschischang, Frey and Loeliger [47]; Loeliger's tutorial [55] is the gentlest introduction, and Dellaert and Kaess [17] show them used for large sparse estimation problems in robotics, which is the closest engineering use to this book's. Directed and undirected graphical models, their factorizations and the relations between them are treated in full by Koller and Friedman [45] and Lauritzen [49]; the point of §2.2 that a directed model can represent a relation only after an orientation has been chosen, and that different orientations encode different factorizations of the same joint, is standard there.

The directionality argument of §2.2.2 is the book's, but its ingredients are not. That a system of equations admits an order of forward substitution exactly when its bipartite equation-variable graph has a matching whose induced dependency graph is acyclic, and that a looped system decomposes into irreducible blocks, is the Dulmage–Mendelsohn decomposition [20]; equation-based modeling languages use it, together with Pantelides' algorithm [68], to decide how to solve a model, and Benveniste, Caillaud and Malandain [5] give the theory. The chapter's contribution is to connect that decomposition to the choice of graphical language: a directed graph is available exactly when the decomposition has no block larger than one.

The bipartite graph of a residual system as the sparsity pattern of its Jacobian, and elimination on it as a graph operation, are the standard graph view of sparse direct methods [16, 78]. The identification of the Markov graph with the pattern of $\mathbf{J}^{\mathsf{T}}\mathbf{J}$ is the same observation as the graph of the normal equations in least-squares theory. The insistence that a physical junction is several variables and several factors is the book's, and it matters because the alternative, one variable per junction and one factor per pipe, is the model that a reader arriving from the graphical-model side would draw first.

Water distribution networks have been drawn as factor graphs in exactly the sense of §2.3. Han, Eguchi, Mehrotra and Venkatasubramanian [33] write the mass balance at each junction and the Hazen–Williams head loss along each pipe as factors, with the heads and the pipe flows as variables, and estimate the state of a damaged network from a few pressure and flow sensors; Irofti, Romero-Ben, Stoican and Puig [35] chain two such graphs, one estimating the leak-free state over a window of readings and one placing a leak from the residuals of the first, and compare the result with interpolation and Kalman-filter baselines on published networks. Both build on the sparse nonlinear least-squares machinery that Dellaert and Kaess [17] describe for robotics. The point of §2.3 that a physical node is several variables and several factors is visible in both: a junction contributes a head, a demand and a balance factor, a pipe a flow and a closure factor, and a sensor a unary factor on one of them.

Summary

- The factor graph of a residual system is its Jacobian's sparsity pattern drawn as a bipartite graph: variables as circles, rows as squares.

- Physical nodes become balance factors; physical edges become a flow variable and a closure factor; reservoirs become constants or unary factors; sensors will become unary factors. A junction is not a variable.
- Of the three graphical languages, the directed one encodes an order of computation that physics does not supply and that loops forbid; the undirected one merges distinct relations into cliques; the factor graph keeps both the relations and their lack of direction.
- An order of computation exists exactly when every irreducible block has one row. The chain has one order; the triangle has three candidate assignments, all cyclic; the cooling loop has a ten-row mass block and an energy problem that is nearly an order.
- Two domains are two layers of one factor graph, coupled through the variables the carrier closures read.
- A physically tree-shaped system can have a loopy factor graph. Counting physical loops does not predict the factor graph's shape.
- The factor graph is not unique. Eliminating variables and merging factors gives coarser factorizations of the same solution set, and the choice governs the cost of inference.
- The Markov graph of the physics is the sparsity pattern of $\mathbf{J}^{\mathsf{T}}\mathbf{J}$, which Part II will identify as the posterior precision.

Exercises

Exercise 2.1. Draw the factor graph of the mixing junction: two inlet streams with known mass rates and enthalpies feeding one junction with one outlet, balanced in both energy and mass, with a carrier closure $P = \dot{m}\, h$ on each stream. Five unknowns, five rows. Identify every cycle.

Exercise 2.2. Complete the argument of §2.2.2 for the water loop: enumerate every assignment of an output variable to each of the five rows such that no two rows share an output, and show that every such assignment contains a directed cycle. How many assignments are there?

Exercise 2.3. A branched service zone is a tree of n junctions fed from a metered inlet, with one pipe into each junction and a known demand at each. Show that its residual system admits an order of computation, write it down, and show that it is the order in which a water engineer sizes the zone by hand, working from the ends inwards. Then open a single valve that closes one loop and show that the order is destroyed.

Exercise 2.4. For the pumping chain, form the reduced two-row system in (h_1, h_2) by eliminating the flows, and confirm by direct substitution that its solutions coincide with those of the four-row system. Draw its factor graph before and after merging the two factors. Which of the three graphs, four-row, two-row, or merged, has the largest factor scope?

Exercise 2.5. Draw the Markov graph of the water loop in row-per-factor form, and verify that it is the sparsity pattern of $\mathbf{J}^{\mathsf{T}}\mathbf{J}$ for the Jacobian of Example 1.2. Then draw the Markov graph of the nodal form (Figure 2.5). Which pairs of variables are adjacent in the first and absent from the second, and why does that not change the solution?

Solutions to selected exercises

Exercise 2.2. The rows and their scopes are b_2 on $\{m_{12}, m_{23}\}$, b_3 on $\{m_{13}, m_{23}\}$, c_{12} on $\{h_2, m_{12}\}$, c_{13} on $\{h_3, m_{13}\}$ and c_{23} on $\{h_2, h_3, m_{23}\}$. An assignment gives each row one of its

variables with no repeats. Since h_2 can be delivered only by c_{12} or c_{23} and h_3 only by c_{13} or c_{23}, and c_{23} can deliver only one of them, at least one of c_{12}, c_{13} delivers its head. Enumerating: (1) $c_{12} \to h_2$, $c_{13} \to h_3$, $c_{23} \to m_{23}$, $b_2 \to m_{12}$, $b_3 \to m_{13}$; (2) $c_{12} \to m_{12}$, $c_{23} \to h_2$, $c_{13} \to h_3$, $b_2 \to m_{23}$, $b_3 \to m_{13}$; (3) the mirror image of (2) with the roles of junctions 2 and 3 exchanged. There are three, and no others. Each has a directed cycle. In (1), c_{12} needs m_{12}, which b_2 delivers from m_{23}, which c_{23} delivers from h_2, which c_{12} was to deliver: $m_{12} \to h_2 \to m_{23} \to m_{12}$. In (2), b_2 delivers m_{23} from m_{12}, c_{12} delivers m_{12} from h_2, and c_{23} delivers h_2 from m_{23} and h_3: again $m_{23} \to h_2 \to m_{12} \to m_{23}$. Case (3) is the mirror. The five rows are one irreducible block, as Proposition 2.1 says, and the script confirms the three assignments and the absence of an acyclic one.

Exercise 2.4. Substitute $m_{12} = k(h_1 - h_2)$ from c_{12} into b_1 and $m_{2a} = k_2(h_2 - h_a)$ from c_{2a} together with m_{12} into b_2:

$$k_1 = k\,(h_1 - h_2) - G, \qquad k_2 = k_2\,(h_2 - h_a) - k\,(h_1 - h_2).$$

Setting $k_1 = 0$ gives $h_1 - h_2 = G/k$; substituting into $k_2 = 0$ gives $h_2 = h_a + G/k_2$; and then $h_1 = h_a + G/k_2 + G/k$, which is the four-row solution, 70 and 90 degrees with the numbers of Example 1.1, and the flows follow from the closures. The two-row factor graph has two variables and two factors, each reading both: a four-cycle, cycle rank one. Merging the factors into one on $\{h_1, h_2\}$ leaves a tree of three nodes, cycle rank zero. The largest scope is three in the four-row form (c_{12} reads h_1, h_2, m_{12}), two in the two-row form, and two in the merged form; the reduced forms have smaller scopes but a denser Markov graph on the variables that remain, which is the trade that Chapter 5 will price.

Part II

Probability on a physical graph

Chapter 3

Where probability enters

The factor graph of Chapter 2 was drawn with indicator factors: each square is one where its row holds and zero where it does not. This chapter keeps every circle and every square and changes what the squares mean. The change is small to state and large in consequence. A square no longer says "this relation holds". It says "this relation holds to within so much", and the "so much" is a number the modeler supplies. Two new kinds of square appear, priors and sensors, both of degree one. The product of all the squares is then a function over the whole state of the system, large where every relation nearly holds and small elsewhere, and after normalization it is a probability distribution. Every later chapter is about that distribution.

The reader who has not worked with probability on continuous quantities will find the necessary definitions in §3.1. They are few. The reader who has may skip to §3.2.

3.1 A probability on the state

Chapter 1 gathered every unknown of the model into a vector $\mathbf{z} \in \mathbb{R}^n$. The deterministic model picked out one point of $\mathbb{R}^n$, the solution of $\mathbf{F}(\mathbf{z}) = \mathbf{0}$. The probabilistic model instead assigns to every point of $\mathbb{R}^n$ a non-negative number, its *density* $p(\mathbf{z})$, with the density integrating to one over the whole space. The density is large at states the model considers likely and small at states it considers unlikely. A region of state space has probability equal to the integral of p over it.

Three operations on a density are used throughout the book.

Marginalization. The density of some components of $\mathbf{z}$ alone, ignoring the others, is obtained by integrating the others out: $p(z_1) = \int p(z_1, z_2)\,\mathrm{d}z_2$. The marginal of the source G in a model of the pumping chain is what one would report as "the source is G, give or take".

Conditioning. The density of some components given that others are known is the joint divided by the marginal of the known ones: $p(z_1 \mid z_2) = p(z_1, z_2)/p(z_2)$. Conditioning on a sensor reading is how a measurement changes what is believed about everything else.

Factorization. A density may be a product of local pieces, $p(\mathbf{z}) \propto \prod_f \varphi_f(\mathbf{z}_f)$, each reading a few components. This is the factor graph of Chapter 2 with the factors now non-negative functions instead of indicators. The proportionality sign hides a constant, the *normalizer* $Z = \int \prod_f \varphi_f \,\mathrm{d}\mathbf{z}$, that makes the integral one.

The Gaussian density is used so often that its form should be recognized on sight. In one dimension,

$$p(z) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(z-\mu)^2}{2\sigma^2}\right), \qquad \text{written } z \sim \mathcal{N}(\mu, \sigma^2), \tag{3.1}$$

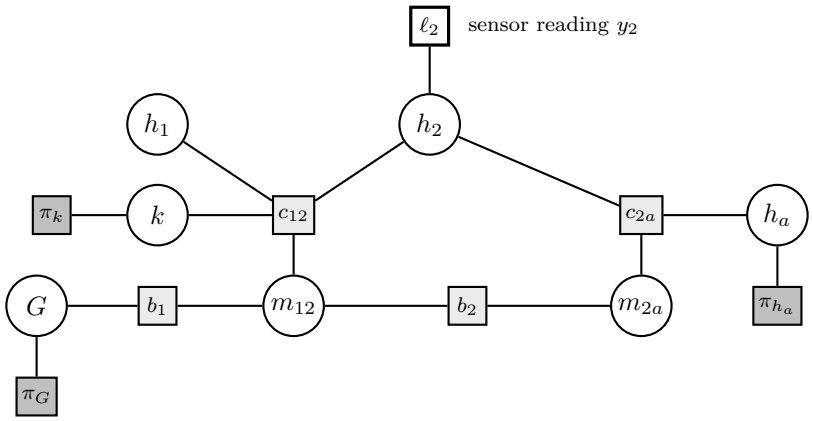


Figure 3.1: The pumping chain with probability attached. Compared with Figure 2.1 three variables have been freed, the source G, the trunk main h_a and the conductance k, and each has acquired a prior (dark squares π). A sensor on h_2 has been added (open square ℓ_2). The four physics factors are the same squares as before, now soft. No edge of the earlier graph has moved.

has mean μ and standard deviation σ. About two thirds of its probability lies within σ of the mean and about 95 percent within 2σ. Its negative logarithm is a quadratic in z, which is the property that makes everything in Chapter 6 work. Appendix B collects the identities the book needs.

One remark on what the numbers mean. A density on the state of a physical system can be read as a frequency, the fraction of days on which the demand at junction 2 takes each value, or as a metre of belief, how strongly the modeler expects the conductance to take each value given a data sheet. The book uses both readings without apology. The machinery is the same, and the practical question is always the same: what does the modeler know, and what does the modeler not know, about each quantity in the model? The answer is written as a factor.

3.2 Five kinds of factor

Every square in the graph is one of five kinds. Two of them, priors and sensors, are new. Three of them, conservation, closure and failure, are the rows of Chapter 1 read in a new way. Figure 3.1 draws all five on the pumping chain and the triangle; the sections that follow define each.

3.2.1 Priors

A *prior* is a unary factor on a quantity the modeler is uncertain about before any measurement is taken: a parameter, a boundary condition, a source. It is what the data sheet, the forecast, or the history says.

The conductance k of the rising main in Example 1.1 is taken from the as-built record as 5 kg/s per metre of head, with a tolerance the record does not state but that experience with pipes of its age puts near twenty percent. Written as a factor,

$$\pi_k(k) = \exp\left(-\frac{(k-5)^2}{2 \cdot 1^2}\right), \tag{3.2}$$

which is $k \sim \mathcal{N}(5, 1^2)$ up to the constant. The trunk main head is a forecast, $h_a \sim \mathcal{N}(20, 3^2)$. The station's delivery is read off a pump curve with a spread, $G \sim \mathcal{N}(100, 20^2)$. Each is a square of degree one attached to its variable (Figure 3.1).

Two things distinguish a prior from a boundary condition. A boundary condition is imposed exactly; a prior is imposed softly, and its width is a statement about how far the true value may be from the nominal one. And a boundary condition removes its variable from the unknowns; a prior leaves it in, so that the measurements can move it. The reservoir of Chapter 1, drawn as a unary factor in Table 2.1, is a prior with zero width. Chapter 2 promised that a reservoir becomes a prior by changing the meaning of one square, and this is the change.

Gaussian priors are the default, not the rule. A parameter that must be positive, a load that is bounded, a component whose availability is one or zero, all need other shapes, and Chapter 7 supplies them. For the present chapter everything is Gaussian.

3.2.2 Closure factors

A closure row $r_k(\mathbf{z}) = 0$ becomes the factor

$$\varphi_k(\mathbf{z}_k) = \exp\left(-\frac{r_k(\mathbf{z}_k)^2}{2\sigma_k^2}\right). \tag{3.3}$$

It is one where the relation holds exactly and falls off as the residual grows, on a scale σ_k in the units of the residual. The scope is unchanged: the factor reads exactly the variables the row read.

The scale σ_k is a statement about the closure, not about the system. A flow law is a model of a physical path. The pipe has a roughness that has changed since it was laid, a head loss that is not quite linear in the flow, and bends, fittings and part-open valves whose minor losses the law ignores. The flow that actually passes differs from $k(h_1 - h_2)$ by an amount the modeler does not know but can bound, and σ_k is that bound. Setting σ_k small says the law is trusted; setting it large says the law is a rough guide. In the limit $\sigma_k \to 0$ the factor becomes the indicator of Chapter 2, and the deterministic model is recovered.

3.2.3 Conservation factors

A balance row becomes a factor of the same form,

$$\varphi_n(\mathbf{z}_n) = \exp\left(-\frac{r_n(\mathbf{z}_n)^2}{2\sigma_n^2}\right), \tag{3.4}$$

and here the reader who takes conservation seriously will object. Energy is conserved. There is no model error in a balance row. Why give it a width at all?

Two answers, one practical and one structural. The practical answer is that a balance row is exact for the system as modeled, and the system as modeled may be wrong: an unmodeled leak, a bypass, a load that is drawing from somewhere the graph does not show. A balance that cannot be satisfied by any state is the signature of exactly such a defect, and a model that forbids violation cannot report it. A balance with a small width can. The size of the violation the posterior settles on is the measurement of the defect, and Chapter 12 makes it a diagnostic tool. The structural answer is that a product of factors with zero width is not a density on $\mathbb{R}^n$; it is concentrated on a lower-dimensional surface, and integrating it, conditioning it, marginalizing

it all require more care than the general machinery provides. Chapter 4 is about that surface, and about what the width does to it. For now: every physics row, balance or closure, is given a width, the widths of balance rows are chosen small, and the deterministic solve is the limit in which all of them go to zero.

3.2.4 Measurement factors

A sensor reads a variable, or a function of several, with an error. The reading is y_i; the variable is z_{o_i}; the error is $\nu_i \sim \mathcal{N}(0, \sigma_i^2)$. Since $y_i = z_{o_i} + \nu_i$, the density of the reading given the state is a Gaussian centered on the variable, and read as a function of the state for the fixed reading it is the *likelihood* factor

$$\ell_i(z_{o_i}) = \exp\left(-\frac{(y_i - z_{o_i})^2}{2\sigma_i^2}\right). \tag{3.5}$$

It is a unary factor on the observed variable, drawn as the open square in Figure 3.1. The sensor scale σ_i is the sensor's accuracy, which its data sheet does state.

Three variations cover what real instrumentation does. A sensor that reads a combination, such as a meter on the total flow leaving a district, has a factor whose scope is every variable in the combination. A sensor with a bias has the bias as a variable of its own, with a prior, in the scope of the factor; the bias is then estimated along with everything else. A sensor that reads a quantity not in $\mathbf{z}$, such as the pressure at a hydrant tapping that the model relates to a node head by a further closure, needs that closure added as a factor and the junction's head added as a variable. In every case the change is local: a square, and perhaps a circle, added to the graph, nothing moved.

This is the promise of §2.7 kept. The deterministic model of Chapter 1 had no place for an observation. The probabilistic model has one, and it is a square of degree one.

3.2.5 Failure factors

Some uncertainty is not about how much but about whether. A pipe is open or it is shut — a valve closed for a repair, a burst isolated, a valve that a crew left shut and did not record. A sensor is working or has failed. Such a quantity is a variable with two values, $a \in \{0, 1\}$, its prior is a probability of each, $\mathbb{P}(a = 1) = 1 - q$ with q the failure rate, and the closure that depends on it reads it in its scope:

$$r_{23}(h_2, h_3, m_{23}, a_{23}) = m_{23} - a_{23}\, k\, (h_2 - h_3). \tag{3.6}$$

With $a_{23} = 1$ this is the pipe closure of Example 1.2. With $a_{23} = 0$ it says the flow is zero whatever the heads: the pipe is shut. The unrecorded closed valve is, in the leak-detection literature, one of the commonest reasons a calibrated hydraulic model stops matching its network, and Chapter 22 shows what a wrong parameter of exactly this kind does to an otherwise sound inference. Figure 3.2 draws it. The variable is discrete, the closure's scope has grown by one, and the product of factors is now a density in the continuous variables times a probability mass in the discrete one. Everything in this chapter still applies; what changes in the computation is the subject of Chapter 7.

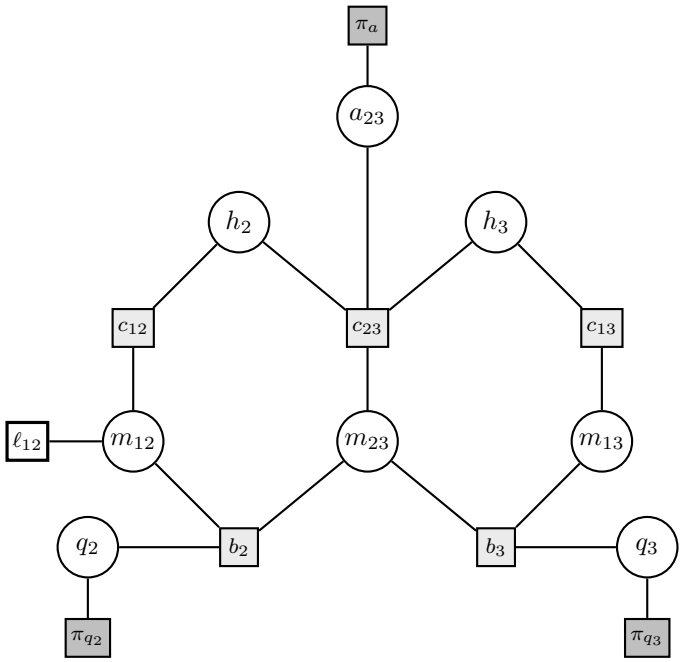


Figure 3.2: The water loop with probability attached. The demands q_2, q_3 are freed and given priors; the line from junction 2 to junction 3 has an availability variable a_{23} with a failure prior, in the scope of its closure; a meter ℓ_{12} reads the flow on pipe 1–2. The structure of Figure 2.3 is intact inside it.

3.2.6 The complete list

Table 3.1 summarizes. The joint distribution of the whole state is their product,

$$\boxed{p(\mathbf{z} \mid \mathbf{y}) \;\propto\; \prod_j \pi_j(z_j) \prod_k \varphi_k(\mathbf{z}_k) \prod_i \ell_i(z_{o_i})} \tag{3.7}$$

with the failure priors folded into the first product. It is written conditional on the readings $\mathbf{y}$ because the likelihoods depend on them; for fixed readings it is a function of $\mathbf{z}$ alone, and it is called the *posterior*. The word means "after the measurements". The same product with the likelihoods removed is the *prior predictive* distribution, what the model believes about the state before any sensor is read.

Principle 3.1. *Probability describes uncertainty around quantities. Physics factors enforce relations between them. The joint distribution is the product of both, and its factor graph is the factor graph of the physics with unary squares added.*

3.3 The joint as an objective

Take the negative logarithm of equation (3.7). Every factor is an exponential of a negative quadratic, so the logarithm is a sum of quadratics:

$$-\log p(\mathbf{z} \mid \mathbf{y}) \;=\; \sum_j \frac{(z_j - \mu_j)^2}{2\sigma_j^2} + \sum_k \frac{r_k(\mathbf{z}_k)^2}{2\sigma_k^2} + \sum_i \frac{(y_i - z_{o_i})^2}{2\sigma_i^2} + \text{const.} \tag{3.8}$$

Table 3.1: The five kinds of factor. The three physics kinds are the rows of Chapter 1; the other two are new.

Kind	Form	Scope	Where it comes from
Prior π	$\exp(-(z-\mu)^2/2\sigma^2)$	one variable	data sheet, forecast, history
Closure φ	$\exp(-r_k(\mathbf{z}_k)^2/2\sigma_k^2)$	the row's variables	a constitutive law and its error
Conservation φ	$\exp(-r_n(\mathbf{z}_n)^2/2\sigma_n^2)$	the row's variables	a balance law; width small
Likelihood ℓ	$\exp(-(y-z)^2/2\sigma_y^2)$	the observed variable	a sensor and its accuracy
Failure π_a	$\mathbb{P}(a)$ on $\{0,1\}$	one discrete variable	a failure rate

The most probable state is the one that minimizes this sum. Read the three terms. The first penalizes departure of each uncertain input from its nominal value, in units of that input's spread. The second penalizes violation of each physics row, in units of that row's width. The third penalizes misfit between each sensor and the variable it reads, in units of the sensor's accuracy. The most probable state is the compromise: it moves the inputs away from nominal as little as it can, violates the physics as little as it can, and misfits the sensors as little as it can, with each "as little" weighted by how much the modeler trusts that term.

Equation (3.8) is a weighted least-squares problem. When every row is linear in $\mathbf{z}$ it is a quadratic and its minimizer solves a linear system; when rows are nonlinear it is solved by the same Newton iteration that solved $\mathbf{F}(\mathbf{z}) = \mathbf{0}$, with the extra terms included. Chapter 6 develops this fully. The point to take now is that the deterministic solve of Chapter 1 is the special case of equation (3.8) in which there are no priors, no sensors, and every $\sigma_k \to 0$: the minimum is then zero, reached only where every row holds, which is the solution.

3.3.1 Residuals in units of their widths

Each term of equation (3.8) is half the square of a *standardized residual*: the departure of an input from its nominal value, the violation of a physics row, or the misfit of a sensor, each divided by its own width. Standardizing puts kg/s, metres and per-unit on one scale, the scale of "how many widths", and it is in that scale that every diagnostic of the book is read. At the most probable state the standardized residuals say where the compromise was struck. For the chain of §3.4 after its first reading, the source sits 0.18 prior widths above nominal, the trunk main 0.06 above, and the sensor is fitted to a hundredth of its accuracy; the sum of the squares, 0.037, is twice the objective at its minimum. After the second reading the four residuals are 0.23, −0.09, 0.08 and −0.07, and the sum is 0.073: two readings, both fitted to a tenth of an accuracy, at the cost of moving the source a quarter of a width. Chapter 4 names this sum the *misfit* and says what value an adequate model should give; Chapter 12 makes it a test. The point to take now is that a posterior is not only a set of estimates but a set of standardized residuals, and the residuals are where the model says whether it was comfortable.

3.3.2 Conditioning a Gaussian on one reading

The worked examples that follow use one operation over and over: a Gaussian belief about a few inputs, and a sensor that reads a linear combination of them. The operation has a closed form, and it is worth deriving once, because every later chapter uses it.

Proposition 3.1 (One linear reading)**.** *Let* $\mathbf{z} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ *with* $\boldsymbol{\Sigma}$ *positive definite, and let a sensor read* $y = \mathbf{h}^{\mathsf{T}}\mathbf{z} + \nu$ *with* $\nu \sim \mathcal{N}(0, \sigma^2)$ *independent of* $\mathbf{z}$*. Then the posterior of* $\mathbf{z}$ *given* y *is*

Gaussian with

$$\boldsymbol{\mu}' = \boldsymbol{\mu} + \mathbf{k}\,(y - \mathbf{h}^{\mathsf{T}}\boldsymbol{\mu}), \qquad \boldsymbol{\Sigma}' = \boldsymbol{\Sigma} - \mathbf{k}\,\mathbf{h}^{\mathsf{T}}\boldsymbol{\Sigma}, \qquad \mathbf{k} = \frac{\boldsymbol{\Sigma}\mathbf{h}}{\mathbf{h}^{\mathsf{T}}\boldsymbol{\Sigma}\mathbf{h} + \sigma^2}. \tag{3.9}$$

Proof. Write $e = y - \mathbf{h}^{\mathsf{T}}\boldsymbol{\mu}$ for the *innovation*, what was read less what the prior predicts, and $s = \mathbf{h}^{\mathsf{T}}\boldsymbol{\Sigma}\mathbf{h} + \sigma^2$ for its variance, so that $\mathbf{k} = \boldsymbol{\Sigma}\mathbf{h}/s$. The proof has four steps.

The negative log posterior. By Bayes' rule $p(\mathbf{z} \mid y) = p(\mathbf{z})\,p(y \mid \mathbf{z})/p(y)$, and $p(y)$ does not depend on $\mathbf{z}$. The prior density is $(2\pi)^{-n/2}\,|\boldsymbol{\Sigma}|^{-1/2}\exp[-\frac{1}{2}(\mathbf{z}-\boldsymbol{\mu})^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{z}-\boldsymbol{\mu})]$. Given $\mathbf{z}$, the number $\mathbf{h}^{\mathsf{T}}\mathbf{z}$ is fixed and ν is still $\mathcal{N}(0,\sigma^2)$, because it is independent of $\mathbf{z}$; so $y \mid \mathbf{z} \sim \mathcal{N}(\mathbf{h}^{\mathsf{T}}\mathbf{z}, \sigma^2)$, with density $(2\pi\sigma^2)^{-1/2}\exp[-(y-\mathbf{h}^{\mathsf{T}}\mathbf{z})^2/2\sigma^2]$. Multiplying the two densities and taking the negative logarithm,

$$-\log p(\mathbf{z} \mid y) = J(\mathbf{z}) + K, \qquad J(\mathbf{z}) = \tfrac{1}{2}(\mathbf{z}-\boldsymbol{\mu})^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{z}-\boldsymbol{\mu}) + \frac{(y-\mathbf{h}^{\mathsf{T}}\mathbf{z})^2}{2\sigma^2},$$

where K collects $\frac{n}{2}\log 2\pi$, $\frac{1}{2}\log|\boldsymbol{\Sigma}|$, $\frac{1}{2}\log 2\pi\sigma^2$ and $\log p(y)$, none of which depends on $\mathbf{z}$. The posterior is proportional to $\exp[-J(\mathbf{z})]$, so its shape in $\mathbf{z}$ is fixed by J; K only makes it integrate to one.

Completing the square. Measure $\mathbf{z}$ from the prior mean, $\mathbf{x} = \mathbf{z} - \boldsymbol{\mu}$. Then $y - \mathbf{h}^{\mathsf{T}}\mathbf{z} = e - \mathbf{h}^{\mathsf{T}}\mathbf{x}$, and expanding the square with $(\mathbf{h}^{\mathsf{T}}\mathbf{x})^2 = \mathbf{x}^{\mathsf{T}}\mathbf{h}\mathbf{h}^{\mathsf{T}}\mathbf{x}$,

$$J = \tfrac{1}{2}\,\mathbf{x}^{\mathsf{T}}\mathbf{P}\,\mathbf{x} - \frac{e}{\sigma^2}\,\mathbf{h}^{\mathsf{T}}\mathbf{x} + \frac{e^2}{2\sigma^2}, \qquad \mathbf{P} = \boldsymbol{\Sigma}^{-1} + \frac{\mathbf{h}\mathbf{h}^{\mathsf{T}}}{\sigma^2}.$$

The matrix $\mathbf{P}$ is positive definite, because $\boldsymbol{\Sigma}^{-1}$ is and $\mathbf{h}\mathbf{h}^{\mathsf{T}}$ is positive semidefinite. Let $\mathbf{m}$ solve $\mathbf{P}\mathbf{m} = \mathbf{h}\,e/\sigma^2$. Expanding the right-hand side of

$$J = \tfrac{1}{2}(\mathbf{x}-\mathbf{m})^{\mathsf{T}}\mathbf{P}\,(\mathbf{x}-\mathbf{m}) + \text{const}$$

recovers the line above. A density proportional to $\exp[-\frac{1}{2}(\mathbf{x}-\mathbf{m})^{\mathsf{T}}\mathbf{P}(\mathbf{x}-\mathbf{m})]$ is the Gaussian with mean $\mathbf{m}$ and covariance $\mathbf{P}^{-1}$. So the posterior of $\mathbf{z} = \boldsymbol{\mu} + \mathbf{x}$ is Gaussian with covariance $\boldsymbol{\Sigma}' = \mathbf{P}^{-1}$ and mean $\boldsymbol{\mu}' = \boldsymbol{\mu} + \mathbf{P}^{-1}\mathbf{h}\,e/\sigma^2$.

The covariance. The claim is $\mathbf{P}^{-1} = \boldsymbol{\Sigma} - \boldsymbol{\Sigma}\mathbf{h}\mathbf{h}^{\mathsf{T}}\boldsymbol{\Sigma}/s$. Multiplying, and using that $\mathbf{h}^{\mathsf{T}}\boldsymbol{\Sigma}\mathbf{h}$ is a number,

$$\mathbf{P}\Big(\boldsymbol{\Sigma} - \frac{\boldsymbol{\Sigma}\mathbf{h}\mathbf{h}^{\mathsf{T}}\boldsymbol{\Sigma}}{s}\Big) = \mathbf{I} + \mathbf{h}\mathbf{h}^{\mathsf{T}}\boldsymbol{\Sigma}\Big(\frac{1}{\sigma^2} - \frac{1}{s} - \frac{\mathbf{h}^{\mathsf{T}}\boldsymbol{\Sigma}\mathbf{h}}{\sigma^2 s}\Big) = \mathbf{I},$$

because the bracket is $(s - \sigma^2 - \mathbf{h}^{\mathsf{T}}\boldsymbol{\Sigma}\mathbf{h})/(\sigma^2 s) = 0$. Since $\boldsymbol{\Sigma}\mathbf{h}\mathbf{h}^{\mathsf{T}}\boldsymbol{\Sigma}/s = \mathbf{k}\,\mathbf{h}^{\mathsf{T}}\boldsymbol{\Sigma}$, this is $\boldsymbol{\Sigma}'$ as stated. The identity is the Sherman–Morrison formula for a rank-one change of $\boldsymbol{\Sigma}^{-1}$.

The mean. From the covariance, $\boldsymbol{\Sigma}'\mathbf{h} = \boldsymbol{\Sigma}\mathbf{h} - \boldsymbol{\Sigma}\mathbf{h}\,(\mathbf{h}^{\mathsf{T}}\boldsymbol{\Sigma}\mathbf{h})/s = \boldsymbol{\Sigma}\mathbf{h}\,\sigma^2/s$. Hence $\boldsymbol{\Sigma}'\mathbf{h}/\sigma^2 = \boldsymbol{\Sigma}\mathbf{h}/s = \mathbf{k}$, and $\boldsymbol{\mu}' = \boldsymbol{\mu} + \mathbf{k}\,e$. □

In one variable, with $\mathbf{h} = 1$, the gain is $\boldsymbol{\Sigma}/(\boldsymbol{\Sigma} + \sigma^2)$ and the covariance formula reads $1/\boldsymbol{\Sigma}' = 1/\boldsymbol{\Sigma} + 1/\sigma^2$: the precisions add. The formulas (3.9) never invert $\boldsymbol{\Sigma}$, and they hold for a singular $\boldsymbol{\Sigma}$ as well, for instance when one input is known exactly; the proof above needs the inverse only to write the prior's density, and Proposition B.3 in Appendix B proves the formulas without it.

What h is, and where the physics is. The vector $\mathbf{h}$ has one entry for each unknown in $\mathbf{z}$, and the sensor reads the weighted sum $\mathbf{h}^\mathsf{T}\mathbf{z} = h_1 z_1 + \cdots + h_n z_n$. A sensor attached directly to the jth unknown has $\mathbf{h}$ equal to the jth unit vector, a one in position j and zeros elsewhere. A sensor attached to a quantity that is not itself in $\mathbf{z}$ reads the combination of the unknowns that determines that quantity, and the physics is what says which combination. The proposition contains no physics. It is the conditioning of any Gaussian vector on any linear reading, and it would hold for prices or pixels. The physics enters through what the proposition is given: which quantities are the unknowns, what the prior says about them, and above all $\mathbf{h}$, the route by which each unknown reaches what the sensor sees. In the example of §3.4 the sensor sits on a head, an entry of $\mathbf{z}$, so its $\mathbf{h}$ is a unit vector; the physics is exact and linear, so every entry of $\mathbf{z}$ is a combination of the source and the trunk main, and the same sensor becomes a vector over those two inputs. Soft physics fits the same mould. A linear row $r(\mathbf{z}) = \mathbf{a}^\mathsf{T}\mathbf{z} - b$ of width σ_k contributes the factor $\exp[-(b - \mathbf{a}^\mathsf{T}\mathbf{z})^2/2\sigma_k^2]$, which is exactly the likelihood of a sensor with $\mathbf{h} = \mathbf{a}$ that reads the value b with accuracy σ_k. A soft physics row is a reading that always reports that the balance holds, and the proposition folds it in like any other. A nonlinear row, or a sensor on a nonlinear function of $\mathbf{z}$, is linearized at the current state, and $\mathbf{h}$ becomes a row of the Jacobian; Chapter 6 takes that step.

Three things are visible in the formula. The mean moves by the *innovation* $y - \mathbf{h}^\mathsf{T}\boldsymbol{\mu}$, the difference between what was read and what was predicted, times a gain $\mathbf{k}$ that is large in the directions the prior is wide and the sensor is accurate. The covariance shrinks by a rank-one amount, in the direction $\boldsymbol{\Sigma}\mathbf{h}$ and in no other: one reading narrows one combination of the inputs and leaves the orthogonal combinations exactly as they were. And the shrinkage does not depend on the reading's value, only on where the sensor sits and how accurate it is, so the posterior spread can be computed before the sensor is read, which is how a sensor's worth is judged before it is bought. Chapter 11 makes this operation the unit of sequential inference.

Several readings at once are handled by the same algebra in a form that is more convenient when there are many.

Proposition 3.2 (Information form)**.** *Write the Gaussian prior by its* precision $\boldsymbol{\Lambda} = \boldsymbol{\Sigma}^{-1}$ *and* information vector $\boldsymbol{\eta} = \boldsymbol{\Lambda}\boldsymbol{\mu}$*. Readings* $\mathbf{y} = \mathbf{H}\mathbf{z} + \boldsymbol{\nu}$ *with* $\boldsymbol{\nu} \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$ *give a posterior with precision and information*

$$\boldsymbol{\Lambda}' = \boldsymbol{\Lambda} + \mathbf{H}^\mathsf{T}\mathbf{R}^{-1}\mathbf{H}, \qquad \boldsymbol{\eta}' = \boldsymbol{\eta} + \mathbf{H}^\mathsf{T}\mathbf{R}^{-1}\mathbf{y}, \tag{3.10}$$

and mean $\boldsymbol{\mu}' = \boldsymbol{\Lambda}'^{-1}\boldsymbol{\eta}'$*, covariance* $\boldsymbol{\Sigma}' = \boldsymbol{\Lambda}'^{-1}$*.*

Proof. The negative log posterior is $\frac{1}{2}\mathbf{z}^\mathsf{T}\boldsymbol{\Lambda}\mathbf{z} - \boldsymbol{\eta}^\mathsf{T}\mathbf{z} + \frac{1}{2}(\mathbf{y} - \mathbf{H}\mathbf{z})^\mathsf{T}\mathbf{R}^{-1}(\mathbf{y} - \mathbf{H}\mathbf{z})$ plus constants; collecting the quadratic and linear terms in $\mathbf{z}$ gives $\frac{1}{2}\mathbf{z}^\mathsf{T}\boldsymbol{\Lambda}'\mathbf{z} - \boldsymbol{\eta}'^\mathsf{T}\mathbf{z}$ with $\boldsymbol{\Lambda}'$ and $\boldsymbol{\eta}'$ as stated, and a Gaussian with that exponent has the mean and covariance claimed. □

In this form a reading *adds* to the precision and the information, and the addition is a sum over readings, one rank-one term each, so readings can be folded in one at a time or all at once with the same result. Proposition 3.1 is the same update expressed in the covariance, the two being related by the matrix identity used in its proof. The information form is the one the rest of the book computes in, because the physics factors of equation (3.7) add to the precision in exactly the same way and keep it sparse; Chapter 6 builds the whole posterior this way.

3.4 A worked example: one sensor, then two

The pumping chain, with numbers, shows what the joint distribution does when a sensor is read. The model is the one of Example 1.1, with the same conductances, $k = 5$ and $k_2 = 2$ kg/s per metre, and two changes. The source and the trunk main are no longer known: they carry the priors $G \sim \mathcal{N}(100, 20^2)$ kg/s and $h_a \sim \mathcal{N}(20, 3^2)$ metres, centred on the values Example 1.1 used. And a sensor on h_2 with accuracy $\sigma_y = 0.5\,\mathrm{m}$ reads $y_2 = 72.0$. To keep the calculation by hand, the closures and balances are taken as exact ($\sigma_k \to 0$). Every number below is produced by a short script, `ch03_chain_posterior.py` in the book's `scripts` directory, which the reader can run.

The example in vectors. The unknown vector is the one of Example 1.1, flows first, with the two inputs appended because they are now unknown too:

$$\mathbf{z} = \begin{pmatrix} m_{12} \\ m_{2a} \\ h_1 \\ h_2 \\ G \\ h_a \end{pmatrix} = \begin{pmatrix} \mathbf{m} \\ \mathbf{h} \\ \mathbf{u} \end{pmatrix}, \qquad \mathbf{u} = \begin{pmatrix} G \\ h_a \end{pmatrix}.$$

Exact physics leaves no freedom beyond the inputs. Example 1.1 solved $\mathbf{F}(\mathbf{z}) = \mathbf{0}$ for the flows and heads in terms of G and h_a: both flows equal G, and $h_1 = \mathbf{h}_1^{\mathsf{T}}\mathbf{u}$, $h_2 = \mathbf{h}_2^{\mathsf{T}}\mathbf{u}$. Every entry of $\mathbf{z}$ is therefore a fixed linear combination of the two inputs,

$$\mathbf{z} = \mathbf{S}\,\mathbf{u}, \qquad \mathbf{S} = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 0.7 & 1 \\ 0.5 & 1 \\ 1 & 0 \\ 0 & 1 \end{pmatrix},$$

whose third and fourth rows are $\mathbf{h}_1^{\mathsf{T}}$ and $\mathbf{h}_2^{\mathsf{T}}$. The prior sits on the inputs,

$$\boldsymbol{\mu}_u = \begin{pmatrix} 100 \\ 20 \end{pmatrix}, \qquad \boldsymbol{\Sigma}_u = \begin{pmatrix} 20^2 & 0 \\ 0 & 3^2 \end{pmatrix} = \begin{pmatrix} 400 & 0 \\ 0 & 9 \end{pmatrix},$$

where the zeros off the diagonal say that the source and the trunk main are independent a priori. It passes to the whole vector as $\boldsymbol{\mu}_z = \mathbf{S}\boldsymbol{\mu}_u = (100, 100, 90, 70, 100, 20)^{\mathsf{T}}$ and $\boldsymbol{\Sigma}_z = \mathbf{S}\boldsymbol{\Sigma}_u\mathbf{S}^{\mathsf{T}}$. The sensor on h_2 reads the fourth entry of $\mathbf{z}$, so in Proposition 3.1 it is the simplest kind: $\mathbf{h} = \mathbf{e}_4$, a one in the position of h_2 and zeros elsewhere. Because $\mathbf{z} = \mathbf{S}\mathbf{u}$, the same reading is $\mathbf{e}_4^{\mathsf{T}}\mathbf{S}\mathbf{u} = \mathbf{h}_2^{\mathsf{T}}\mathbf{u}$, a reading of the inputs through $\mathbf{h}_2$. The entries of $\mathbf{h}_2$ are the physics. The first, $1/k_2 = 0.5$ metres per kg/s, is how far one kg/s of source raises the junction; the second, 1, is how far one metre of trunk-main head does. (The bold $\mathbf{h}$ is the sensor's vector; the italic h_i are heads.)

The covariance $\boldsymbol{\Sigma}_z$ has rank two, because six unknowns are driven by two inputs, so it is not positive definite. The calculation is therefore done where the freedom is. Proposition 3.1 is applied to $\mathbf{u}$, with $\mathbf{h}_2$, and $\mathbf{S}$ carries the result to all of $\mathbf{z}$: $\boldsymbol{\mu}'_z = \mathbf{S}\boldsymbol{\mu}'_u$ and $\boldsymbol{\Sigma}'_z = \mathbf{S}\boldsymbol{\Sigma}'_u\mathbf{S}^{\mathsf{T}}$. The proposition's formulas applied to $\mathbf{z}$ directly, with $\mathbf{h} = \mathbf{e}_4$ and the rank-two $\boldsymbol{\Sigma}_z$, give the same posterior to twelve digits.

Table 3.2: The pumping chain: prior, posterior after the h_2 sensor, posterior after both sensors. Means and standard deviations; the last two rows are the heads predicted from each distribution.

	Prior	After $y_2 = 72.0$	After y_2 and $y_1 = 93.0$
G [kg/s]	100.0 ± 20.0	103.7 ± 5.8	104.6 ± 2.9
h_a [m]	20.0 ± 3.0	20.2 ± 2.9	19.7 ± 1.7
$\mathrm{corr}(G, h_a)$	0	-0.985	-0.979
h_2 predicted	70.0 ± 10.4	72.0 ± 0.5	72.0 ± 0.4
h_1 predicted	90.0 ± 14.3	92.7 ± 1.3	93.0 ± 0.5

Before the reading. The prior predicts h_2 at $\mathbf{e}_4^{\mathsf{T}}\boldsymbol{\mu}_z = \mathbf{h}_2^{\mathsf{T}}\boldsymbol{\mu}_u = 0.5 \cdot 100 + 1 \cdot 20 = 70.0$, with variance $\mathbf{h}_2^{\mathsf{T}}\boldsymbol{\Sigma}_u\mathbf{h}_2 = 0.5^2 \cdot 400 + 1^2 \cdot 9 = 109$, a spread of 10.44 metres. Most of that variance comes from the source, 100 of the 109, against the trunk main's 9. The same two products with $\mathbf{h}_1$, the third entries of $\boldsymbol{\mu}_z$ and $\boldsymbol{\Sigma}_z$, give the prior for h_1, 90.0 ± 14.3.

After one reading. The reading $y_2 = 72.0$ gives the innovation $e = y_2 - \mathbf{h}_2^{\mathsf{T}}\boldsymbol{\mu}_u = 2.0$. Proposition 3.1 then takes a handful of products:

$$\boldsymbol{\Sigma}_u\mathbf{h}_2 = \begin{pmatrix} 400 \cdot 0.5 \\ 9 \cdot 1 \end{pmatrix} = \begin{pmatrix} 200 \\ 9 \end{pmatrix}, \qquad s = \mathbf{h}_2^{\mathsf{T}}\boldsymbol{\Sigma}_u\mathbf{h}_2 + \sigma^2 = 109 + 0.25 = 109.25,$$

$$\mathbf{k} = \frac{\boldsymbol{\Sigma}_u\mathbf{h}_2}{s} = \begin{pmatrix} 1.8307 \\ 0.0824 \end{pmatrix}, \qquad \boldsymbol{\mu}_u' = \boldsymbol{\mu}_u + \mathbf{k}\,e = \begin{pmatrix} 100 + 3.66 \\ 20 + 0.16 \end{pmatrix} = \begin{pmatrix} 103.66 \\ 20.16 \end{pmatrix},$$

$$\boldsymbol{\Sigma}_u' = \boldsymbol{\Sigma}_u - \mathbf{k}\,(\boldsymbol{\Sigma}_u\mathbf{h}_2)^{\mathsf{T}} = \begin{pmatrix} 400 & 0 \\ 0 & 9 \end{pmatrix} - \begin{pmatrix} 1.8307 \cdot 200 & 1.8307 \cdot 9 \\ 0.0824 \cdot 200 & 0.0824 \cdot 9 \end{pmatrix} = \begin{pmatrix} 33.87 & -16.48 \\ -16.48 & 8.26 \end{pmatrix},$$

where $\mathbf{k}\,\mathbf{h}_2^{\mathsf{T}}\boldsymbol{\Sigma}_u = \mathbf{k}\,(\boldsymbol{\Sigma}_u\mathbf{h}_2)^{\mathsf{T}}$ because $\boldsymbol{\Sigma}_u$ is symmetric. Read the result off the matrices. The diagonal of $\boldsymbol{\Sigma}_u'$ holds the variances: the source's falls from 400 to 33.87, a spread of 5.8 kg/s, and the trunk main's from 9 to 8.26, a spread of 2.9 metres. The off-diagonal entry, zero before, is now -16.48, a correlation of $-16.48/(5.82 \cdot 2.87) = -0.985$. The posterior of the whole vector follows through $\mathbf{S}$:

$$\boldsymbol{\mu}_z' = \mathbf{S}\boldsymbol{\mu}_u' = (103.66,\ 103.66,\ 92.73,\ 72.00,\ 103.66,\ 20.16)^{\mathsf{T}},$$

and the diagonal of $\mathbf{S}\boldsymbol{\Sigma}_u'\mathbf{S}^{\mathsf{T}}$ gives the spreads $(5.82, 5.82, 1.34, 0.50, 5.82, 2.87)$. Both flows sit at 103.7 ± 5.8 kg/s, since each equals the source, and the junction at 72.0 ± 0.50 metres, the sensor's own accuracy. The posterior, in Table 3.2, is exactly this: the source up by 3.7 kg/s and the trunk main by 0.16 metres; the source moves more because it had more room to move. The source's spread falls from 20 to 5.8 kg/s, a factor of three and a half, from a single head reading taken two nodes away. The trunk main's spread barely moves, from 3.0 to 2.9. And the two, independent under the prior, are now correlated at -0.985: the posterior knows that $h_a + 0.5G$ is close to 72, so it knows that if the source is higher the trunk main is lower, without knowing which. Figure 3.3 draws the prior and the two posteriors as one-standard-deviation ellipses in the (G, h_a) plane; the first posterior is the thin diagonal ridge.

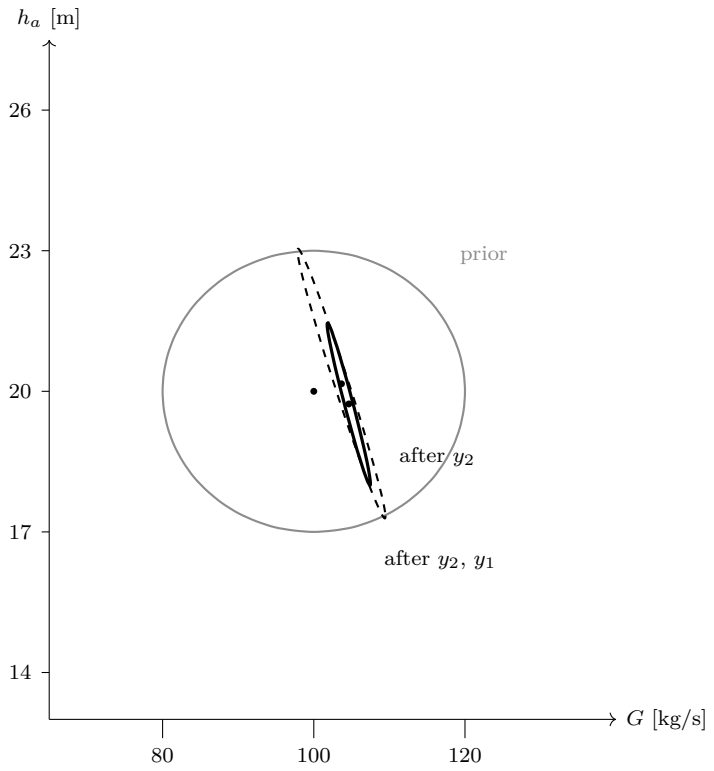


Figure 3.3: One-standard-deviation ellipses of the joint distribution of source and trunk-main head: the prior (grey, axis-aligned, because the two are independent), the posterior after the h_2 reading (dashed, a ridge along $h_a + 0.5G = 72$), and the posterior after both readings (solid, shorter along the ridge). Dots mark the means. Generated by the same script as Table 3.2.

A virtual sensor. h_1 has not been measured, but it is the third entry of $\mathbf{z}$, and the posterior predicts it: $\mathbf{h}_1^\mathsf{T}\boldsymbol{\mu}'_u = 0.7 \cdot 103.66 + 20.16 = 92.7$, with variance $\mathbf{h}_1^\mathsf{T}\boldsymbol{\Sigma}'_u\mathbf{h}_1 = 1.79$, a spread of 1.3. Its prior spread was 14.3. The reading at node 2 has tightened the prediction at node 1 by a factor of ten, through the physics: h_1 and h_2 differ by G/k, and G is now known to ± 5.8. This is the first virtual sensor in the book. Chapter 12 is about them. Figure 3.4 draws the three predictive densities of h_1, before any reading, after the first, and after the second, with the second reading marked; the first reading turned a broad prior into a prediction sharp enough to be tested, and the second passed the test.

A second reading. Now a sensor on h_1 reads $y_1 = 93.0$. The reading falls within the virtual sensor's range, which is the first thing to check: had it read 85, six standard deviations from the prediction, the model rather than the source would be the suspect. Proposition 3.1 applies again, with the first posterior in the place of the prior. The sensor reads the third entry of $\mathbf{z}$, so on $\mathbf{z}$ it is $\mathbf{h} = \mathbf{e}_3$, and on the inputs it is $\mathbf{h}_1$. The innovation is $93.0 - \mathbf{h}_1^\mathsf{T}\boldsymbol{\mu}'_u = 0.27$, and

$$\boldsymbol{\Sigma}'_u\mathbf{h}_1 = \begin{pmatrix} 7.23 \\ -3.28 \end{pmatrix}, \qquad s = \mathbf{h}_1^\mathsf{T}\boldsymbol{\Sigma}'_u\mathbf{h}_1 + \sigma^2 = 1.79 + 0.25 = 2.04, \qquad \mathbf{k} = \frac{\boldsymbol{\Sigma}'_u\mathbf{h}_1}{s} = \begin{pmatrix} 3.55 \\ -1.61 \end{pmatrix},$$

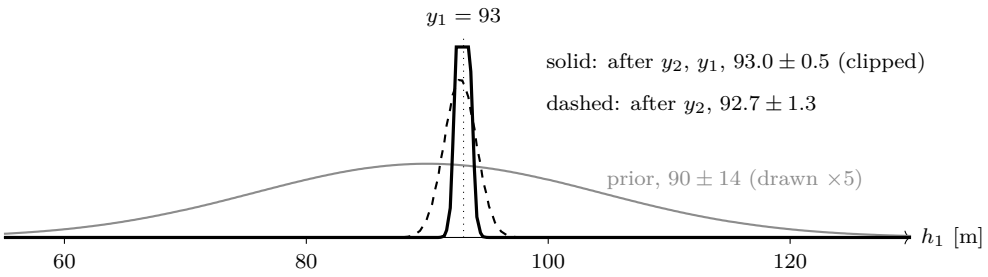


Figure 3.4: The predictive density of the unmeasured discharge head h_1 under the prior (grey), after the junction reading y_2 (dashed), and after both readings (solid, clipped in height). The dotted line is the reading $y_1 = 93.0$ when it arrives: inside the dashed prediction's range, so the model is not surprised.

$$\boldsymbol{\Sigma}_u'' = \boldsymbol{\Sigma}_u' - \mathbf{k}\,(\boldsymbol{\Sigma}_u'\mathbf{h}_1)^{\mathsf{T}} = \begin{pmatrix} 8.20 & -4.85 \\ -4.85 & 2.99 \end{pmatrix}.$$

The gain on the trunk main is negative although $\mathbf{h}_1$ has a positive entry for it. The first reading left the source and the trunk main correlated, so a discharge head above its prediction is blamed on the source, and the correlation pulls the trunk main down. The mean moves by $\mathbf{k}\,e = (0.97, -0.44)$, and the result is the third column of Table 3.2. Proposition 3.2, which folds both readings in at once, gives the same posterior to twelve digits. The source is now 104.6 ± 2.9, the trunk main 19.7 ± 1.7. The second sensor's lever arm on the source is only the difference $h_1 - h_2 = G/k$, with a slope of 0.2 metres per kg/s against two readings each accurate to 0.5 metres, which is why the source improves by a factor of two and not by more. The correlation is still -0.979: two readings of head, both of which are "trunk main plus something times source", cannot fully separate the two. A sensor on the flow, or on the trunk main itself, would. Which sensor to add next is a question the posterior can answer before the sensor is bought, and Chapter 12 returns to it.

3.5 A second example: the triangle with a flow meter

The same operation on the water loop shows what a loop does to it. The model is the one of Example 1.2, with $k = 10$ kg/s per metre and the physics exact, and one change: the two demands are no longer known. They carry the priors $q_2 \sim \mathcal{N}(-1.5, 0.3^2)$ and $q_3 \sim \mathcal{N}(-0.5, 0.3^2)$ kg/s, independent and centred on the second set of demands Example 1.2 used. A demand prior of this width — a fifth of the demand — is what a utility actually has between meter reads, and it is the prior every chapter after this one inherits. Every number is from `ch03_triangle_posterior.py`.

Table 3.3: The water loop: prior, posterior after the meter on pipe 1–2, posterior after a second meter on the flow the trunk main delivers. Demands and flows in kg/s, heads in metres; means and standard deviations.

	prior	after $m_{12} = 1.10$	after m_{12} and boundary 2.05
q_2	-1.500 ± 0.300	-1.421 ± 0.136	-1.273 ± 0.060
q_3	-0.500 ± 0.300	-0.460 ± 0.269	-0.773 ± 0.069
$\mathrm{corr}(q_2, q_3)$	0	-0.976	-0.961
m_{12}	1.167 ± 0.224	1.101 ± 0.020	1.107 ± 0.019
m_{13}	0.833 ± 0.224	0.780 ± 0.135	0.940 ± 0.027
m_{23}	-0.333 ± 0.141	-0.320 ± 0.134	-0.167 ± 0.043
boundary flow	2.000 ± 0.424	1.881 ± 0.139	2.047 ± 0.020

The example in vectors. As for the chain, the unknown vector is the one of Example 1.2, flows first, with the demands appended:

$$\mathbf{z} = \begin{pmatrix} m_{12} \\ m_{13} \\ m_{23} \\ h_2 \\ h_3 \\ q_2 \\ q_3 \end{pmatrix} = \begin{pmatrix} \mathbf{F} \\ \mathbf{h} \\ \mathbf{u} \end{pmatrix}, \qquad \mathbf{u} = \begin{pmatrix} q_2 \\ q_3 \end{pmatrix}.$$

The exact physics, equation (1.22) with the flows from the closures, makes every entry a linear combination of the demands,

$$\mathbf{z} = \mathbf{S}\,\mathbf{u}, \qquad \mathbf{S} = \frac{1}{3}\begin{pmatrix} -2 & -1 \\ -1 & -2 \\ 1 & -1 \\ 2/k & 1/k \\ 1/k & 2/k \\ 3 & 0 \\ 0 & 3 \end{pmatrix},$$

and the flow the trunk main delivers, $m_{12} + m_{13}$, is $-(q_2 + q_3)$. Under the prior the flows are 1.167 ± 0.224, 0.833 ± 0.224 and -0.333 ± 0.141, and the boundary flow 2.000 ± 0.424. A meter on pipe 1–2 reads the first entry of $\mathbf{z}$: on $\mathbf{z}$ it is $\mathbf{h} = \mathbf{e}_1$, and on the demands it is the first row of $\mathbf{S}$, $(-\frac{2}{3}, -\frac{1}{3})^{\mathsf{T}}$. It reads $m_{12} = 1.10$ with accuracy 0.02, a third of a spread below its prediction. As for the chain, Proposition 3.1 is applied to $\mathbf{u}$ and $\mathbf{S}$ carries the result to $\mathbf{z}$; applied to $\mathbf{z}$ directly, with $\mathbf{h} = \mathbf{e}_1$, it gives the same posterior.

Proposition 3.1 gives the posterior of the demands, $\mathbf{S}$ carries it to the flows, Table 3.3 lists both, and Figure 3.5 draws the demands. The meter reads the direction $(-0.894, -0.447)$ of the demand plane, mostly q_2; along it the spread falls from 0.300 to 0.027, and across it, in the direction $(0.447, -0.894)$, it stays at 0.300 exactly. The demands become $q_2 = -1.421 \pm 0.136$ and $q_3 = -0.460 \pm 0.269$, correlated at -0.976: the meter has fixed two thirds of one plus one third of the other, so if one is lower the other must be higher. The unmetered flows move too, through the loop: m_{13} from 0.833 ± 0.224 to 0.780 ± 0.135, m_{23} from -0.333 ± 0.141 to -0.320 ± 0.134,

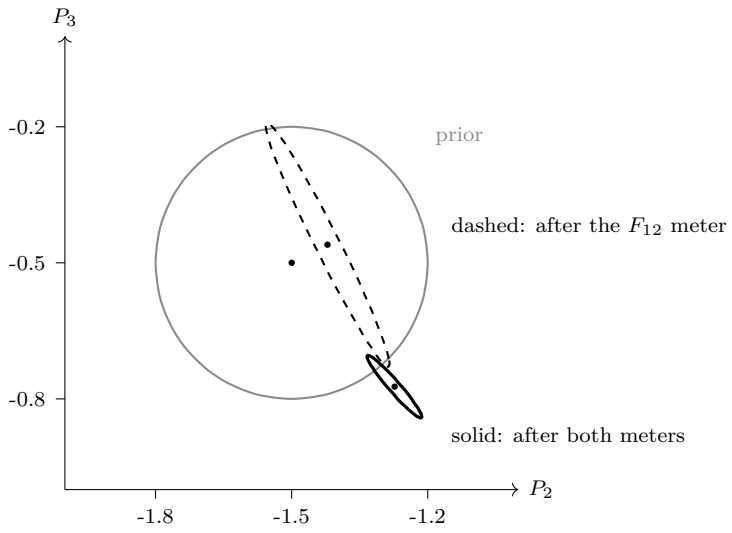


Figure 3.5: One-standard-deviation ellipses of the two demands of the water loop: the prior (grey), the posterior after the meter on pipe 1–2 (dashed), and the posterior after a second meter on the boundary flow (solid). The first meter reads one direction of the demand plane and flattens the ellipse along it only; the second reads a different direction and closes it.

and the flow the trunk main delivers from 2.000 ± 0.424 to 1.881 ± 0.139. One meter on one pipe has narrowed the district's total demand by a factor of three, because the total is mostly what the meter reads. That sentence is the whole argument for district metering, and the rest of the section is about its limits.

A second meter, on the flow the trunk main delivers, reads 2.05 ± 0.02. The boundary flow is not an entry of $\mathbf{z}$ but the sum of two, so on $\mathbf{z}$ this meter is $\mathbf{h} = \mathbf{e}_1 + \mathbf{e}_2$, and on the demands it is the sum of the first two rows of $\mathbf{S}$, the direction $(-1, -1)$. That is not the first meter's direction, and the two together pin both demands: $q_2 = -1.273 \pm 0.060$, $q_3 = -0.773 \pm 0.069$ kg/s, the flow on the pipe between them -0.167 ± 0.043. The prior had put q_3 at -0.5; the two meters have moved it by nearly a full prior spread, to -0.77, because the inlet meter says the total is 2.05 and the pipe meter says most of it is not at junction 2. Neither meter reads q_3; both inform it through the loop. This is the looped twin of the chain's virtual sensor, and the loop is what makes it work: in a branched zone a meter on one street says nothing about the demand on another. It is also why a looped network, which every utility builds for reliability, is worth more to an estimator than a branched one — a point Chapter 21 makes on two hundred and eleven unknowns.

3.6 Reading the graph

The example computed numbers. The graph shows, before any number is computed, why they came out as they did.

The sensor ℓ_2 in Figure 3.1 touches only h_2. The source G is three squares away: through c_{2a} to m_{2a}, through b_2 to m_{12}, through b_1 to G. Yet the reading moved G more than it moved anything else. Information travels along every path in the factor graph, and how much arrives depends on the factors along the way, not on the number of steps. A stiff closure passes

information nearly unattenuated; a wide prior receives it readily. The reading reached G through three exact physics factors and found a prior twenty kg/s wide. It reached h_a through one exact factor and found a prior three metres wide. It moved what was loosest.

The two priors π_G and π_{h_a} share no factor, and under the prior their variables are independent. After the reading they are correlated at -0.985. The reading is a factor on h_2, and h_2 is tied to both. Two variables that are independent become dependent when something they both influence is observed.

The size of the correlation has a closed form, and it is worth deriving because it says when the effect is severe. Let two independent inputs $u_1 \sim \mathcal{N}(\mu_1, s_1^2)$ and $u_2 \sim \mathcal{N}(\mu_2, s_2^2)$ be read together through $y = au_1 + bu_2 + \nu$, $\nu \sim \mathcal{N}(0, \sigma^2)$. With $\mathbf{\Sigma} = \mathrm{diag}(s_1^2, s_2^2)$ and $\mathbf{h} = (a, b)$, Proposition 3.1 gives $\mathbf{\Sigma}' = \mathbf{\Sigma} - \mathbf{\Sigma h h}^{\mathsf{T}}\mathbf{\Sigma}/(a^2 s_1^2 + b^2 s_2^2 + \sigma^2)$, whose off-diagonal entry is $-ab s_1^2 s_2^2 / D$ and whose diagonal entries are $s_1^2(b^2 s_2^2 + \sigma^2)/D$ and $s_2^2(a^2 s_1^2 + \sigma^2)/D$, with D the denominator. The posterior correlation is therefore

$$\rho' \;=\; -\frac{ab\, s_1 s_2}{\sqrt{(a^2 s_1^2 + \sigma^2)(b^2 s_2^2 + \sigma^2)}}. \tag{3.11}$$

It tends to -1 (for $ab > 0$) as the sensor becomes accurate relative to both inputs' contributions, and to zero as the sensor becomes useless; it is large exactly when the sensor reads a combination it can resolve much better than either prior could. For the chain, $a = 0.5$ metres per kg/s, $s_1 = 20$, $b = 1$, $s_2 = 3$, $\sigma = 0.5$: $\rho' = -0.5 \cdot 20 \cdot 3/\sqrt{(100.25)(9.25)} = -0.985$, the table's value. The sensor resolves $h_a + 0.5G$ to half a metre, where the prior knew it to ten; that is a twenty-fold gain along the ridge and none across it, and the correlation is the ridge's aspect ratio. This is the same fact that Chapter 2 met as "factorization is not independence": local factors, global dependence. It has a name in the graphical-model literature, *explaining away*, and in Chapter 5 it is stated precisely as a rule for reading conditional independence off the graph.

> **Principle 3.2.** *A measurement anywhere is information everywhere. It travels along the factor graph, attenuated by tight factors and absorbed by loose ones, and it correlates whatever it cannot separate.*

3.7 In practice

Write down what you know about every input. A prior is not a nuisance to be set vague; it is the record of the data sheet, the forecast and the history, and its width decides how much a reading moves that input. An input given a width of a thousand will absorb every reading; one given a width of zero is a boundary condition.

Give the physics a width and say why. A closure's width is the modeler's bound on the law's error, and it should be justified in those terms: contact resistance, a neglected radiative term, a linear approximation. Balance rows get a small width so that the model can report a violation rather than refuse one; Chapter 4 says how small.

Check the innovation before believing the update. The innovation, reading minus prediction, in units of the predicted spread, is the first number to look at. A reading three spreads from its prediction is either a discovery or a defect, and the posterior will absorb it either way. Chapter 12 turns this into a test.

Ask what a sensor reads before buying it. By Proposition 3.1 the posterior spread after a reading does not depend on the reading. Compute it for every candidate sensor and buy the one that narrows the quantity you care about; a sensor that reads a direction the prior has already pinned buys nothing.

Expect correlations and report them. Two readings of the same kind leave the inputs correlated, as the chain's two heads left G and h_a at -0.98 and the water loop's two meters left the demands at -0.96. A report that gives each input its own error bar and no correlation has thrown away the posterior's most useful content: the combination that is known well and the combination that is not.

Keep the deterministic limit in view. Every posterior in this chapter tends to the deterministic solve as the priors widen and the physics widths vanish. When a posterior surprises, take that limit and see whether the surprise is in the physics or in the widths.

3.8 What is missing

Three things were done in this chapter without justification, and the next three chapters supply it.

The physics rows were softened with widths σ_k, and in the worked example they were then set to zero and the physics eliminated by hand. Chapter 4 says what the product of factors is when the widths are zero, what the widths do to it, and how to choose them.

The graph was read for how information travels and where it correlates. Chapter 5 makes that reading exact: which variables, given which others, are independent, and what a separator is.

The worked example was Gaussian throughout and solved in closed form. Chapter 6 shows that when every row is linear and every factor Gaussian the posterior is Gaussian, that its precision is the weighted normal matrix of the Jacobian with the sparsity of Chapter 2, and that the whole computation is sparse linear algebra. Chapter 7 then asks what survives when a row is nonlinear or a variable is discrete, as a_{23} already is.

Notes and further reading

Attaching priors to the uncertain inputs of a physical model, reading sensors as likelihoods, and asking for the posterior over the physical state is the Bayesian formulation of inverse problems; Stuart [88] gives the mathematical framework and Kaipio and Somersalo [37] the computational one, and both treat the Gaussian case of this chapter's worked example at length. The same objective, weighted least squares over physics residuals and measurement misfits, has two older engineering homes. In power systems it is state estimation, introduced by Schweppe [83] and developed in the books of Monticelli [64] and Abur and Gómez Expósito [1]; in chemical process engineering it is data reconciliation, of which Mah, Stanley and Downing [57] is the classical statement, with conservation constraints, estimation of unmeasured flows and detection of gross errors, all of which reappear in Chapter 12. Both traditions usually take the physics as hard and carry no priors on the inputs; the soft physics and the input priors of this chapter are what let a single formulation cover estimation, diagnosis and the deterministic solve.

The closest precedent for a factor graph over a physical network with message passing on it is Chavali and Nehorai's distributed power-system state estimation [14]. Its scope is one domain and one query; the framework of this book generalizes the graph to multiple conservation domains and the queries to those of Part IV, but the reader should know that the construction of Figure 3.2 has been drawn before.

The remark in §3.1 that a density can be read as a frequency or as a metre of belief, and that the machinery is the same, is the position of MacKay [56], whose book is the best general reference for the reader who wants the probability of this chapter developed with the same practical emphasis. Explaining away, named in §3.6, is Pearl's term [70], and Chapter 5 gives it a precise form for physical graphs.

Summary

- A density on the state assigns a likelihood to every point of $\mathbb{R}^n$. Marginalization integrates out, conditioning divides, factorization multiplies local pieces.
- Five kinds of factor: priors on uncertain inputs, closure and conservation factors with widths, likelihoods from sensors, and failure priors on discrete variables. The physics factors are the rows of Chapter 1; priors and likelihoods are unary squares added to the graph.
- The posterior is the product, equation (3.7). Its negative logarithm is a weighted least-squares objective whose deterministic limit is the solve of Chapter 1.
- One linear reading moves the mean by the innovation times a gain and shrinks the covariance by a rank-one amount in one direction; the shrinkage does not depend on the reading's value.
- One head reading two nodes from an uncertain source cut its spread by a factor of three and a half and predicted an unmeasured head to ± 1.3 metres. The second reading landed inside that prediction.
- On the water loop, one pipe meter narrowed the total demand by a factor of three and moved an unmetered demand through the loop; a second meter in a different direction pinned both.
- Information travels along the graph and is absorbed by the loosest factors; observing a shared effect correlates independent causes.

Exercises

Exercise 3.1. Repeat the worked example with the sensor on h_1 read first and the sensor on h_2 second. Show that the final posterior is the same and that the intermediate one is different. Which single sensor, on h_1 or on h_2, tells more about G, and why?

Exercise 3.2. Add a third sensor to the worked example that reads the trunk main head directly, first with accuracy 1.0 metre and then with 0.3. Compute the posterior in each case and show that the correlation between G and h_a falls in magnitude, to about 0.6 in the second case. Explain in terms of Figure 3.1 why this sensor breaks the ridge and the first two did not.

Exercise 3.3. In the worked example, free the conductance k with the prior $\mathcal{N}(5, 1^2)$ and keep everything else. The map from inputs to h_1 is now nonlinear in (G, k). Write the objective of equation (3.8) for this case, find its minimizer numerically, and compare the minimizer with the second column of Table 3.2.

Exercise 3.4. Give the balance row b_2 of the chain a width σ_b and suppose the two sensors read $y_2 = 72.0$ and $y_1 = 105.0$, inconsistent with any state of the exact model. Show that as $\sigma_b \to 0$ the objective's minimum grows without bound, and that for finite σ_b the minimizer violates the balance by an amount that identifies the size of an unmodeled draw, a leak, at node 2.

Exercise 3.5. For the triangle of Figure 3.2, write the joint distribution as a product over the two values of a_{23}: $p(\mathbf{z}) = \mathbb{P}(a_{23} = 1)\, p(\mathbf{z} \mid a_{23} = 1) + \mathbb{P}(a_{23} = 0)\, p(\mathbf{z} \mid a_{23} = 0)$. Draw the factor graph for each of the two terms and state what has happened to the loop in the second.

Solutions to selected exercises

Exercise 3.1. Reading $y_1 = 93.0$ first gives $G = 104.1 \pm 4.2$ and $h_a = 20.1 \pm 2.9$, correlated at -0.986; reading y_2 second gives $G = 104.6 \pm 2.9$, $h_a = 19.7 \pm 1.7$, correlated at -0.979, which is the third column of Table 3.2 to fourteen digits. The order cannot matter, because the posterior is the product of the prior and both likelihoods, and a product does not depend on the order of its factors; Proposition 3.1 applied twice is one way of computing that product, and the intermediate posterior is the only thing that depends on which factor went in first. The sensor on h_1 alone leaves G at ± 4.2, the sensor on h_2 alone at ± 5.8. The reason is the slope: $h_1 = h_a + 0.7G$ rises seven tenths of a metre per kg/s and $h_2 = h_a + 0.5G$ five tenths, so at equal accuracy the h_1 sensor has the larger lever arm on the source. Both sensors share the same unit slope on h_a, which is why neither separates the two.

Exercise 3.2. With the two head sensors read, a trunk-main sensor of accuracy 1.0 metre reading 19.5 gives $G = 104.9 \pm 1.5$, $h_a = 19.6 \pm 0.9$, and a correlation of -0.924; with accuracy 0.3 it gives $G = 105.0 \pm 0.8$, $h_a = 19.5 \pm 0.3$, and -0.636. The first two sensors both read "trunk main plus a multiple of the source" and so both lie along the ridge $h_a + cG$; in Figure 3.1 they attach to h_2 and h_1, which are tied to h_a and G through the same chain of factors, and the ridge is what that chain cannot distinguish. The trunk main sensor attaches to h_a alone, reads the direction $(0, 1)$ of the (G, h_a) plane, which crosses the ridge, and Proposition 3.1 shrinks the covariance in a direction that no earlier reading had touched. The correlation does not vanish, because the ridge is only partly crossed at accuracy 1.0 and mostly crossed at 0.3; a sensor on G itself, or on a flow, would cross it at right heads.

Chapter 4

Hard constraints, soft constraints, and the manifold

Chapter 3 gave every physics row a width and then, in its worked example, set the widths to zero and eliminated the physics by hand. Both moves were made without justification. This chapter supplies it. It says what the product of factors is when the widths are zero, what the widths do when they are not, why a working model uses finite widths, and how to choose them. The chapter ends with the same pumping chain as before, now with all six unknowns kept and the physics rows soft, and shows the soft posterior converge to the hard one as the widths shrink, and the price paid for shrinking them.

4.1 The hard joint and its manifold

Split the unknowns into two groups. The *inputs* $\mathbf{u}$ are the quantities that carry priors: parameters, boundary conditions, sources. The *solved variables* $\mathbf{x}$ are everything else: potentials, flows, storage states. The physics rows read both, $\mathbf{F}(\mathbf{x}, \mathbf{u}) = \mathbf{0}$, and there are as many rows as solved variables, $m = n_x$, with the Jacobian in the solved variables, $\mathbf{J}_x = \partial\mathbf{F}/\partial\mathbf{x}$, invertible. That is the well-posedness of §1.5 restated: for each value of the inputs the physics determines the solved variables. Write the determined value as $\mathbf{x} = X(\mathbf{u})$. The function X is what a solver computes; it is not available in closed form except for the smallest models, but it exists, and it is smooth wherever $\mathbf{J}_x$ is invertible.

Now take the widths of Chapter 3 to zero. A physics factor $\exp(-r_k^2/2\sigma_k^2)$, divided by its normalizer $\sqrt{2\pi}\sigma_k$, becomes the delta function $\delta(r_k)$: zero away from $r_k = 0$, and with unit integral across it. The product of the m physics factors is $\delta(\mathbf{F}(\mathbf{x}, \mathbf{u}))$, which is zero except on the set

$$\mathcal{M} \;=\; \{(\mathbf{x}, \mathbf{u}) : \mathbf{F}(\mathbf{x}, \mathbf{u}) = \mathbf{0}\} \;=\; \{(X(\mathbf{u}), \mathbf{u})\}, \tag{4.1}$$

the graph of X. This set is the *constraint manifold*. It has dimension n_u, the number of inputs, inside a space of dimension $n_x + n_u$. Every physically admissible state of the system lies on it, and no state off it is admissible.

The hard joint, prior times physics, is therefore not a density on $\mathbb{R}^n$ at all. It is a density on $\mathcal{M}$: a distribution over the inputs, carried along by X to the solved variables. The natural way to write it is

$$p(\mathbf{x}, \mathbf{u}) \;=\; p(\mathbf{u})\; \delta\big(\mathbf{x} - X(\mathbf{u})\big), \tag{4.2}$$

which says: draw the inputs from their prior, solve, and the solved variables are what the solve returns. Its marginal over the inputs is the prior, as it should be, since physics alone says nothing about which inputs are likely. Its marginal over any solved variable is the distribution of that variable under the prior, the *prior predictive*, which the worked example of §3.4 computed for h_2 as 70.0 ± 10.4.

Remark 4.1 (Two delta functions). Equation (4.2) and the product of row factors $p(\mathbf{u})\,\delta(\mathbf{F}(\mathbf{x},\mathbf{u}))$ are not the same function. Integrating the second over $\mathbf{x}$ gives $p(\mathbf{u})/|\det \mathbf{J}_x(X(\mathbf{u}),\mathbf{u})|$, by the change of variables $\mathbf{r} = \mathbf{F}(\mathbf{x},\mathbf{u})$ at fixed $\mathbf{u}$. The two agree when $\mathbf{J}_x$ is constant, which is when the physics is linear in the solved variables, and differ otherwise by a factor that weights inputs according to how compliant the physics is there. The general statement is the coarea formula of geometric measure theory [22, 23]: a product of delta functions of residuals is not invariant under a reparameterization of the residuals unless the corresponding Jacobian factor is carried along, and equation (4.2) is the parameterization-free object. Within one model the factor matters only where the physics is nonlinear, and for the models of this book it is small there. Between two models whose physics rows differ it is neither small nor common: comparing a branch in which a pipe is open with one in which it is out, by the evidence of their soft row factors, weights each by $1/|\det \mathbf{J}_x|$ of its own physics, and on the water loop the two determinants differ by a factor of three. Chapter 7 shows the error this makes and states the correction (Proposition 7.1).

4.2 Conditioning on the manifold

A sensor reads a solved variable x_j with likelihood $\ell(x_j)$. Conditioning the hard joint on the reading multiplies equation (4.2) by $\ell(x_j)$ and, since $x_j = X_j(\mathbf{u})$ on the manifold, gives

$$p(\mathbf{u} \mid y) \;\propto\; p(\mathbf{u})\,\ell(X_j(\mathbf{u})). \tag{4.3}$$

This is the *input-space* form of the posterior. It is a distribution over the inputs alone. The solved variables are recovered from it by X: their posterior is the distribution of $X(\mathbf{u})$ when $\mathbf{u}$ is drawn from equation (4.3).

The worked example of §3.4 was exactly this computation. Its inputs were $\mathbf{u} = (G, h_a)$; its X was the linear map $\mathbf{z} = \mathbf{S}\mathbf{u}$ from the inputs to the whole unknown vector; its posterior was equation (4.3) with Gaussian prior and Gaussian likelihood, which is Gaussian, and Table 3.2 tabulated it. The manifold was a plane in $\mathbb{R}^6$ parameterized by two numbers, and the whole calculation lived on that plane.

For the chain the map is explicit. With inputs $\mathbf{u} = (G, h_a)$ and solved variables $\mathbf{x} = (m_{12}, m_{2a}, h_1, h_2)$, flows first as in Example 1.1,

$$X(\mathbf{u}) = \begin{pmatrix} G \\ G \\ h_a + 0.7\,G \\ h_a + 0.5\,G \end{pmatrix}, \qquad \frac{\partial X}{\partial \mathbf{u}} = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 0.7 & 1 \\ 0.5 & 1 \end{pmatrix}, \tag{4.4}$$

which is the matrix $\mathbf{S}$ of §3.4 without its last two rows, the ones that return the inputs themselves. The derivative is dense: every solved variable depends on the source, and the two heads on both inputs, although no single physics row read more than three variables. Exercise 4.1 asks for the general statement.

The input-space form is exact, and it is always available in principle. It has two costs. The first is that every evaluation of the right-hand side of equation (4.3) at a new $\mathbf{u}$ needs $X(\mathbf{u})$,

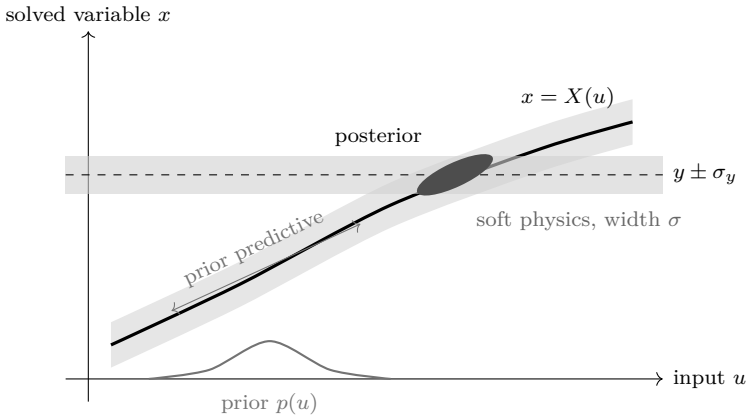


Figure 4.1: One input, one solved variable. The thick curve is the constraint manifold $x = X(u)$; the hard joint lives on it. The grey band around the curve is where the soft joint puts its mass, a distance of order σ from the curve. The prior on u sits on the horizontal axis and is carried up the curve to the prior predictive on x. A sensor reading y is a horizontal band. The posterior is the product: the dark region where the sensor band meets the curve, weighted by how much prior mass lies below it. Softening thickens the curve; it does not move the intersection.

which is a solve of the physics; and every gradient needs $\partial X/\partial \mathbf{u} = -\mathbf{J}_x^{-1}\mathbf{J}_u$, which is a solve per input or an adjoint solve per output. Chapter 14 makes much of this derivative, but it is not free. The second cost is structural. The physics of Chapter 1 was sparse: each row read a few variables. The map X is not. Every solved variable depends, in general, on every input, because information travels along the whole graph, so $\partial X/\partial \mathbf{u}$ is a dense $n_x \times n_u$ matrix and the factor graph of equation (4.3) is a single factor of scope $\mathbf{u}$. Eliminating the physics eliminates the sparsity with it.

4.3 The soft joint

Keep the widths finite. The joint is then

$$p_\sigma(\mathbf{x}, \mathbf{u} \mid y) \;\propto\; p(\mathbf{u})\, \ell(x_j) \prod_{k=1}^{m} \exp\!\left(-\frac{r_k(\mathbf{x}, \mathbf{u})^2}{2\sigma_k^2}\right), \tag{4.5}$$

a density on all of $\mathbb{R}^{n_x+n_u}$, positive everywhere, largest near the manifold and falling off away from it on a scale set by the widths. Figure 4.1 draws the situation for one input and one solved variable.

Three things happen as the widths go to zero, and they are what the chapter promised to establish.

Proposition 4.1 (Zero-width limit)**.** *Let the physics be linear in* $(\mathbf{x}, \mathbf{u})$*, the prior and the likelihood Gaussian, and all widths* $\sigma_k = \sigma$*. Then the soft posterior* (4.5) *is Gaussian for every* $\sigma > 0$*, and as* $\sigma \to 0$*:*

1. *its marginal over the inputs converges to the hard posterior* (4.3)*;*
2. *its mean converges to the point of* $\mathcal{M}$ *that minimizes the prior and likelihood terms of equation* (3.8) *subject to* $\mathbf{F}(\mathbf{x}, \mathbf{u}) = \mathbf{0}$*;*

3. *the posterior standard deviation of each physics residual r_k is at most σ, so the mass off the manifold vanishes with the width.*

Proof. Write the linear physics as $\mathbf{F} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$ with $\mathbf{A}$ square and invertible, so that $X(\mathbf{u}) = -\mathbf{A}^{-1}\mathbf{B}\mathbf{u}$. The soft joint is the exponential of a negative quadratic in $(\mathbf{x}, \mathbf{u})$, hence Gaussian for every $\sigma > 0$.

For item 1, change variables from $(\mathbf{x}, \mathbf{u})$ to $(\mathbf{r}, \mathbf{u})$ with $\mathbf{r} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$; the map is linear with constant Jacobian $\det \mathbf{A}$, so densities transform by a constant. In the new variables $\mathbf{x} = X(\mathbf{u}) + \mathbf{A}^{-1}\mathbf{r}$ and the soft joint is $p(\mathbf{u})\,\ell(X_j(\mathbf{u}) + (\mathbf{A}^{-1}\mathbf{r})_j)\,\exp(-\|\mathbf{r}\|^2/2\sigma^2)$. The last factor, normalized, is a Gaussian on $\mathbf{r}$ of spread σ about zero; integrating $\mathbf{r}$ out gives $p(\mathbf{u})$ times the average of $\ell(X_j(\mathbf{u}) + \epsilon)$ over $\epsilon \sim \mathcal{N}(0, \sigma^2\|(\mathbf{A}^{-1})_{j\cdot}\|^2)$, which for the Gaussian ℓ is a Gaussian in $X_j(\mathbf{u})$ whose spread is the sensor's with $\sigma^2\|(\mathbf{A}^{-1})_{j\cdot}\|^2$ added in quadrature. As $\sigma \to 0$ this tends to $\ell(X_j(\mathbf{u}))$, and the marginal to the hard posterior (4.3).

For item 2, let $\mathbf{z}_\sigma$ minimize $\Phi_\sigma(\mathbf{z}) = \Phi_0(\mathbf{z}) + \|\mathbf{F}(\mathbf{z})\|^2/2\sigma^2$, with Φ_0 the prior and likelihood terms, and let $\mathbf{z}^\star$ minimize Φ_0 subject to $\mathbf{F} = \mathbf{0}$. Since $\mathbf{z}^\star$ is feasible, $\Phi_\sigma(\mathbf{z}_\sigma) \le \Phi_\sigma(\mathbf{z}^\star) = \Phi_0(\mathbf{z}^\star)$, so $\|\mathbf{F}(\mathbf{z}_\sigma)\|^2 \le 2\sigma^2(\Phi_0(\mathbf{z}^\star) - \Phi_0(\mathbf{z}_\sigma))$, which tends to zero because Φ_0 is bounded below. Every limit point of $\mathbf{z}_\sigma$ is therefore feasible, and has $\Phi_0 \le \Phi_0(\mathbf{z}^\star)$ by the same inequality, so it is a constrained minimizer; for a strictly convex Φ_0 the minimizer is unique and $\mathbf{z}_\sigma \to \mathbf{z}^\star$.

For item 3, the posterior precision contains the term $\mathbf{h}_k\mathbf{h}_k^\mathsf{T}/\sigma^2$ for the residual $r_k = \mathbf{h}_k^\mathsf{T}\mathbf{z}$, and the remaining terms are positive semidefinite. Adding a positive semidefinite matrix to a precision cannot increase any variance, so the posterior variance of r_k is at most its variance under the precision $\mathbf{h}_k\mathbf{h}_k^\mathsf{T}/\sigma^2$ alone, restricted to the direction $\mathbf{h}_k$, which is σ^2; Exercise 4.2 asks for the one-line matrix argument. □

For nonlinear physics the same statements hold locally, with the volume factor of Remark 4.1 appearing in the first.

The proposition is what licenses the two shortcuts of Chapter 3. The worked example eliminated the physics and computed on the manifold; that is the $\sigma \to 0$ limit of the soft model, and §4.6 confirms it numerically. And the objective of equation (3.8) was described as having the deterministic solve as its limit; that is item 2 with no prior and no sensor.

4.4 Why soften

If the hard model is exact and the soft model converges to it, why not use the hard model? Three reasons, in increasing order of importance.

The soft model is an unconstrained problem. Minimizing the prior and sensor terms subject to $\mathbf{F} = \mathbf{0}$ is a constrained problem, and its optimality conditions are a saddle-point system in the unknowns and m Lagrange multipliers. Minimizing equation (4.5)'s negative logarithm is an unconstrained least-squares problem whose optimality conditions are a positive definite linear system, or, for nonlinear physics, a sequence of them. The second is what the Newton iteration of Chapter 1 already solves, with more rows. The width is the reciprocal square root of a penalty weight, and the soft model is the quadratic penalty method for the constrained one. That method is old, and its limitation is known: the penalty weight cannot be made arbitrarily large, because the conditioning of the linear system degrades with it. §4.5 returns to this.

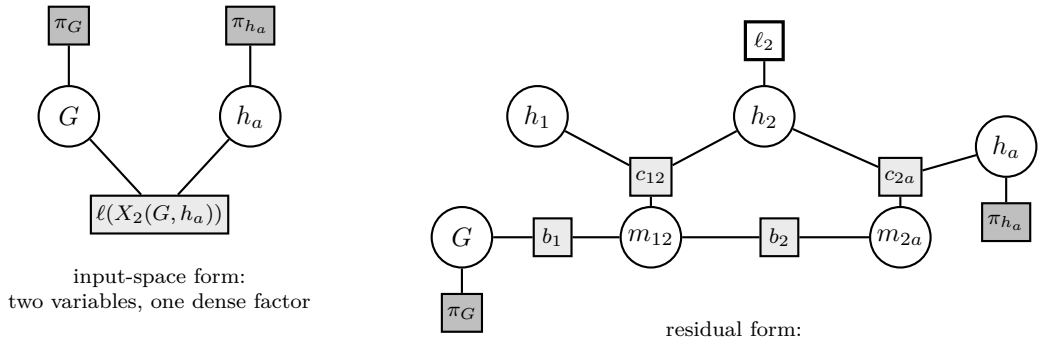


Figure 4.2: The same posterior in two forms. Left: the input-space form of §4.2, in which the physics has been solved and the likelihood is one factor reading every input. Right: the residual form of the soft joint, in which the physics rows remain as factors of small scope and the sensor reads one variable. On six variables the difference is cosmetic; on a network it is the difference between a dense matrix on the inputs and a sparse one on everything.

The soft model keeps the sparsity. This is the structural reason, and Figure 4.2 draws it. The factor graph of equation (4.5) is the factor graph of Chapter 3: physics squares of small scope, unary priors and sensors. Its precision matrix, which Chapter 6 constructs, has the sparsity of the Markov graph of §2.5, linear in the size of the model. The hard model in input-space form has thrown that away: a single dense factor on the inputs. For six unknowns the difference is nothing. For a network of ten thousand nodes it is the difference between a sparse factorization that takes a fraction of a second and a dense one that does not fit in memory. Every decomposition method of Part III, and every scaling argument of Part V, works on the soft graph.

The soft model can be wrong in a way that shows. This is the reason that matters most in practice. The hard model asserts that the state lies on $\mathcal{M}$. If the sensors report something that no point of $\mathcal{M}$ can produce, the hard model has no answer: the likelihood is zero everywhere on the manifold, and the posterior is undefined. A real system produces such readings routinely, because the model is missing something: a leak, a bypass, a load the graph does not show, a sensor that has drifted. The soft model always has an answer. It places its posterior off the manifold, by an amount that trades the physics widths against the prior and sensor widths, and the residuals r_k at that posterior say which rows were violated and by how much. Those residuals are the diagnostic. A model that forbids violation cannot report it; a model that permits it, at a cost, reports both the violation and the cost. Chapter 12 builds fault detection on exactly this, and §4.6 shows a first instance.

> **Principle 4.1.** *Physics is softened not because it is uncertain but because the model of it is. A finite width keeps the problem unconstrained, keeps the graph sparse, and lets a violated law be seen and measured rather than forbidden.*

4.5 Choosing the widths

The widths are numbers the modeler supplies, and there are four sources of them.

Table 4.1: Where the widths come from, with the values used in the book's examples. Relative widths are fractions of the row's scale after the equilibration described in the text.

factor	width is	source	examples in the book
sensor	instrument accuracy	data sheet, calibration	0.5 m gauge; 0.02 kg/s meter
prior	spread of the input	data sheet, forecast, history	source ±20 kg/s; demand ±0.3 kg/s; room ±3 K
closure	bound on the law's error	neglected terms, handbook scatter	10^{-3} relative when the law is trusted
balance	admissible violation	model completeness, numerics	10^{-3} relative; loosened to find a leak

Sensor widths come from the instrument. A thermocouple's data sheet states its accuracy; a power meter's states its class. These are the least negotiable numbers in the model, and the worked examples use them as given.

Prior widths come from data sheets, forecasts, and history, as §3.2 said. They are statements about the modeler's ignorance and should be honest: a conductance known to twenty percent gets a twenty percent width, not five, and a forecast whose past errors had a spread of three metres gets three metres.

Closure widths come from what is known about the law's error. A linearized law that neglects a term of known size gets a width of that size. A correlation from a handbook with a stated scatter gets that scatter. A law believed exact, such as a linearized pipe at a fixed conductance, gets a small width, but not zero, for the reasons of the previous section.

Balance widths are the small ones. A balance is exact for the system as modeled; its width is there to admit violation when the model is incomplete, and to keep the problem unconstrained. It should be small compared with the flows it balances, and no smaller than the numerics allow. Table 4.1 collects the four sources with the values the book's examples use.

That last clause has a definite content. The physics rows have units, a balance in kg/s, a closure in kg/s or metres or per-unit, and a width must be stated in the row's units. It is convenient, and in a working implementation necessary, to scale each row so that its typical magnitude is of order one, and then to state widths as fractions of that scale. Call the fraction the *relative width*. With rows so scaled, the precision matrix of the soft model has entries of order one from the priors and sensors and of order $1/\rho^2$ from the physics at relative width ρ, and its condition number grows as $1/\rho^2$. The connection between condition number and lost accuracy is a standard result of numerical analysis [34]: a backward-stable solve loses about as many decimal digits as the condition number has, so in double precision a condition number near 10^{10} leaves six digits and one near 10^{14} leaves two. The widths that follow from this are implementation heuristics, not universal constants; they depend on the row scaling, the solver and the precision. With rows equilibrated as described, a relative width of 10^{-3} for the balance rows is a comfortable choice, 10^{-4} is near the edge, and 10^{-5} fails on models of moderate size. The next section shows the growth on the chain, and a reader with a different normalization should repeat the sweep on their own model.

Failure modes. A width too large lets the posterior satisfy the sensors by breaking the physics rather than by moving an input, and the estimates of the inputs stay near their priors when the data should have moved them. A width too small does the opposite: the posterior moves inputs to implausible values, and blames sensors, rather than admit a violation that the model should have allowed. Neither failure is silent. Both show in the *misfit*: the sum of squared standardized residuals at the posterior over every factor, physics, prior and sensor, which is twice the objective (3.8) there. Chapter 12 turns the misfit into a test, and the statement behind the test is worth deriving here, because it is what gives the number a scale.

Proposition 4.2 (The misfit of an adequate model)**.** *Let a model have m linear Gaussian factors, each of the form $\mathbf{h}_k^\mathsf{T}\mathbf{z} - c_k \sim \mathcal{N}(0, \sigma_k^2)$ with independent errors, on $n < m$ unknowns, with the stacked standardized rows $\mathbf{W}^{1/2}\mathbf{J}$ of full column rank. If the data were generated by the model, the misfit at the posterior mean, the sum of the squared standardized residuals, is a chi-squared variable with $m - n$ metres of freedom; its expectation is $m - n$ and its standard deviation $\sqrt{2(m-n)}$.*

Proof. Standardize every factor: let $\mathbf{a}_k = \mathbf{h}_k/\sigma_k$ and $b_k = c_k/\sigma_k$, so that the model says $\mathbf{A}\mathbf{z} = \mathbf{b} + \boldsymbol{\epsilon}$ with $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_m)$ and $\mathbf{A}$ the $m \times n$ matrix of rows $\mathbf{a}_k^\mathsf{T}$. The posterior mean minimizes $\|\mathbf{A}\mathbf{z} - \mathbf{b}\|^2$, so it is the least-squares solution $\hat{\mathbf{z}} = (\mathbf{A}^\mathsf{T}\mathbf{A})^{-1}\mathbf{A}^\mathsf{T}\mathbf{b}$, and the standardized residual vector is $\mathbf{b} - \mathbf{A}\hat{\mathbf{z}} = (\mathbf{I} - \mathbf{P})\mathbf{b}$ with $\mathbf{P} = \mathbf{A}(\mathbf{A}^\mathsf{T}\mathbf{A})^{-1}\mathbf{A}^\mathsf{T}$ the orthogonal projector onto the column space of $\mathbf{A}$. If the data came from the model, $\mathbf{b} = \mathbf{A}\mathbf{z}_{\text{true}} - \boldsymbol{\epsilon}$, and since $(\mathbf{I} - \mathbf{P})\mathbf{A} = \mathbf{0}$ the residual is $-(\mathbf{I} - \mathbf{P})\boldsymbol{\epsilon}$: the true state has dropped out. The misfit is $\boldsymbol{\epsilon}^\mathsf{T}(\mathbf{I} - \mathbf{P})\boldsymbol{\epsilon}$, a standard Gaussian vector projected onto the orthogonal complement of an n-dimensional subspace, which is the sum of $m - n$ independent squared standard normals: chi-squared with $m - n$ degrees of freedom. □

The proposition carries its assumptions: linear rows, Gaussian errors with the stated widths, independence. For nonlinear rows it holds locally; for widths that are wrong it fails in a way that is itself informative, since a misfit systematically below $m - n$ says the widths are too wide and one systematically above says they are too narrow or the model is missing something. A value far above $m - n$, by more than a few multiples of $\sqrt{2(m-n)}$, says the model, the widths, or the data are inconsistent. The statement is not a property of an arbitrary factor graph, and the example below shows one failure of each kind.

4.6 Worked example: the chain with soft physics

Take the pumping chain of Figure 3.1 with all six unknowns, $\mathbf{z} = (m_{12}, m_{2a}, h_1, h_2, G, h_a)$, the priors $G \sim \mathcal{N}(100, 20^2)$ and $h_a \sim \mathcal{N}(20, 3^2)$, the conductances $k = 5$ and $k_2 = 2$ known, and the four physics rows each given the same width σ in kg/s. Everything is linear, so the posterior is Gaussian and is computed by the script `ch04_soft_physics.py` in the book's `scripts` directory; every number below is from its output.

Convergence. With the single sensor $y_2 = 72.0$ of §3.4, Table 4.2 sweeps the width from 10 kg/s down to 1 milliwatt. At $\sigma = 10$, a tenth of the flow, the physics is loose: the source's posterior spread is 13.5, the correlation with the trunk main is only -0.25, and the largest physics residual at the posterior mean is half a kg/s. The reading has been partly absorbed by bending the physics rather than by moving the inputs. At $\sigma = 1$ the estimates are within a few

Table 4.2: The chain with soft physics, one sensor $y_2 = 72.0$: posterior of the source and room as the physics width shrinks. The last column is the condition number of the posterior precision matrix. Hard-physics reference from Table 3.2: $G = 103.66 \pm 5.82$, $h_a = \pm 2.87$, correlation -0.985.

σ [W]	mean G	sd G	sd h_a	corr	max $\lvert r_k \rvert$ [W]	cond
10	102.17	13.52	2.93	-0.247	5.4×10^{-1}	1.6×10^{3}
1	103.64	6.03	2.87	-0.944	9.1×10^{-3}	6.4×10^{3}
0.1	103.66	5.82	2.87	-0.985	9.2×10^{-5}	5.9×10^{5}
0.01	103.66	5.82	2.87	-0.985	9.2×10^{-7}	5.9×10^{7}
0.001	103.66	5.82	2.87	-0.985	9.2×10^{-9}	5.9×10^{9}

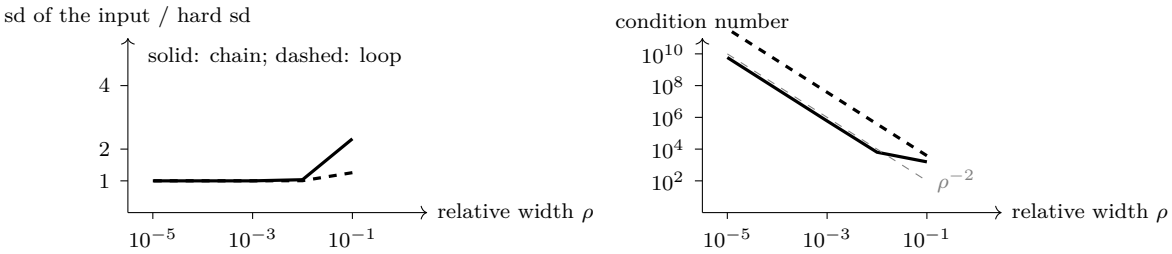


Figure 4.3: The width sweep on the chain (solid) and on the water loop of §4.7 (dashed), against the relative width, the physics width divided by the scale of the flows (100 kg/s for the chain, one kg/s for the loop). Left: the posterior spread of the uncertain input relative to its hard-physics value; both reach the hard value at a relative width of 10^{-3}. Right: the condition number of the posterior precision, which grows as the inverse square of the relative width on both models.

percent of the hard values. At $\sigma = 0.1$ and below they agree with Chapter 3's hard posterior to every digit shown, the residuals are below a tenth of a milliwatt, and the posterior spread of each residual equals the width, as Proposition 4.1 says. The soft model has become the hard one.

The last column is the price, and Figure 4.3 draws it together with the same sweep on the water loop of §4.7. Each factor of ten in the width costs a factor of a hundred in the condition number, as §4.5 predicted. At a milliwatt, which is a relative width of 10^{-5} against flows of a hundred kg/s, the condition number is 6×10^{9} on a six-variable model; a larger model with the same relative width would be past the edge. A width of 0.01 to 0.1 kg/s gives the hard answer at a condition number a solver does not notice.

An inconsistent reading. Now keep every width at 0.01 kg/s and read both sensors, $y_2 = 72.0$ as before and $y_1 = 105.0$. In the exact model $h_1 - h_2 = G/k$, so the two readings imply $G = 165$ kg/s; but then $m_{2a} = 165$ and $h_2 = h_a + 82.5$, which with $h_2 = 72$ puts the trunk main near -10 metres. No state of the exact model is consistent with both readings and both priors. The soft model with tight physics has to answer anyway, and its answer is in the first row of Table 4.3: source 147 kg/s, room 0.4 metres, the h_2 sensor missed by two metres and the h_1 sensor by one and a half. Every physics residual is zero. The model has kept its laws and thrown away its priors and its sensors: the trunk main sits six and a half of its prior spreads below the forecast, and each sensor is off by three or four of its accuracies. The misfit at this posterior, the sum of squared standardized residuals over all eight factors, is 74; for an adequate model with eight

Table 4.3: Inconsistent readings $y_2 = 72.0$, $y_1 = 105.0$: posterior as the width of the node-2 balance is loosened while every other row stays at 0.01 kg/s. The leak estimate is the violation of that balance at the posterior mean. Last column: misfit at the posterior; about 2 is expected of an adequate model.

σ_{b_2} [W]	leak [W]	G [W]	h_a [m]	h_2, h_1 fitted	misfit
0.01	0.0 ± 0.0	147.2 ± 2.9	0.4 ± 1.7	74.05, 103.49	74.0
1	1.1 ± 1.0	147.5 ± 2.9	0.8 ± 1.8	74.02, 103.52	72.8
10	38.1 ± 5.9	157.5 ± 3.3	13.1 ± 2.6	72.85, 104.34	32.6
100	58.3 ± 7.3	162.9 ± 3.5	19.9 ± 3.0	72.21, 104.80	10.6

factors and six unknowns it would be about 2. The model is telling the reader, loudly, that something is wrong, and it is telling the reader the wrong thing about what.

Suppose the modeler suspects the balance at node 2: the junction may be losing heat somewhere the graph does not show, to the mounting bolts say. Loosen that one row and leave the others tight. The remaining rows of Table 4.3 show what happens. At a width of one kg/s nothing changes; the row is still too stiff to absorb sixty kg/s. At ten kg/s the posterior has moved: the balance is violated by 38 kg/s, the trunk main has come back to 13 metres, the sensors are fitted to within a metre. At a hundred kg/s the posterior is coherent: the balance at node 2 is violated by 58 ± 7 kg/s, the source is 163 kg/s, the trunk main is 19.9 metres, within a tenth of its forecast, and both sensors are fitted to a fifth of a metre. The misfit has fallen from 74 to 10.6. It is still above 2, and the reason is visible in the table: the source is three of its prior spreads above nominal. That is either a second thing wrong or a prior that was too narrow, and the modeler now knows to ask.

Read the residual of the loosened row as a number. The balance at node 2 said heat in equals heat out. At the posterior, heat in exceeds heat out by 58 kg/s. That is the water leaving the junction by a path the model does not contain, and the model has estimated it, with an error bar, from two head readings, without being told a leak exists. The row was violated, and the violation is the measurement.

For comparison, the consistent readings of §3.4, $y_2 = 72.0$ and $y_1 = 93.0$, through the same soft model with every width at 0.01 kg/s, give $G = 104.6 \pm 2.9$ and $h_a = 19.7 \pm 1.7$, identical to the hard result, with a misfit of 0.07. An adequate model with consistent data sits near the bottom of the misfit's range; an inadequate one does not. Figure 4.4 draws the loosening sweep, beside the same sweep on the water loop that follows.

4.7 A second example: the loop with an unmodeled leak

The water loop of §3.5 carries the same lessons onto a graph with a cycle in it, and here the "leak" of the chain's analogy is a leak. Keep all seven unknowns of §3.5, flows first: the three flows, the two heads and the two demands. Give the five physics rows a common width σ in kg/s, the demands their priors -1.5 ± 0.3 and -0.5 ± 0.3 kg/s, and read meters of accuracy 0.02 kg/s. Every number is from `ch04_triangle_soft.py`.

Convergence. With the single meter $m_{12} = 1.10$ of Chapter 3, Table 4.4 sweeps the width from 0.1 kg/s, a tenth of the flows, down to 10^{-5}. At 0.1 the posterior is loose, the correlation

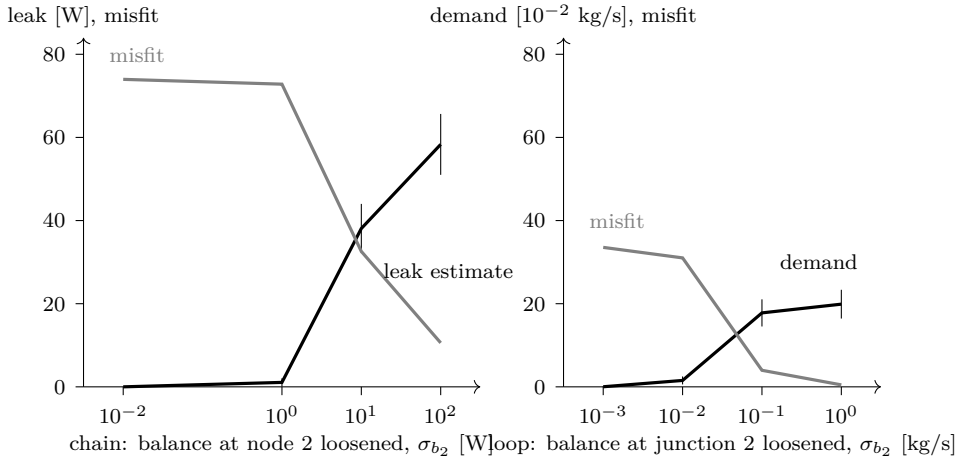


Figure 4.4: Loosening one balance row against inconsistent readings. Left: the chain, the balance at node 2 loosened; the leak estimate with its error bar (black) rises from nothing to 58 ± 7 kg/s as the width passes ten kg/s, and the misfit (grey) falls from 74 toward the value an adequate model would give. Right: the water loop of §4.7, the balance at junction 2 loosened; the unmodeled leak rises to 0.20 kg/s and the misfit falls from 34 to below one. In both, the row that is allowed to be violated is the row that reports the defect.

Table 4.4: The water loop with soft physics, one meter $m_{12} = 1.10$ kg/s: posterior of the demands as the physics width shrinks. Hard reference from Table 3.3: $q_2 = -1.421 \pm 0.136$, $q_3 = -0.460 \pm 0.269$ kg/s, correlation -0.976.

σ [kg/s]	q_2	q_3	corr	max $\lvert r_k \rvert$	cond
0.1	-1.433 ± 0.171	-0.466 ± 0.274	-0.649	7.5×10^{-3}	3.8×10^{3}
0.01	-1.421 ± 0.137	-0.460 ± 0.269	-0.971	8.8×10^{-5}	3.8×10^{5}
0.001	-1.421 ± 0.136	-0.460 ± 0.269	-0.976	8.8×10^{-7}	3.8×10^{7}
10^{-4}	-1.421 ± 0.136	-0.460 ± 0.269	-0.976	8.8×10^{-9}	3.8×10^{9}
10^{-5}	-1.421 ± 0.136	-0.460 ± 0.269	-0.976	8.8×10^{-11}	3.8×10^{11}

between the demands only -0.65, and the physics residuals are of order 10^{-2}: the meter has been partly absorbed by bending the loop rather than by moving the demands. At 0.01 the demands are within a thousandth of the hard values and at 0.001 they agree to every digit shown, with the correlation -0.976 of Table 3.3. The condition number grows by a hundred per decade, exactly as on the chain, and Figure 4.3 shows the two models' curves lying on top of each other once the width is expressed relative to the flows. A relative width of 10^{-3} is the comfortable choice on both.

Inconsistent readings. Now read three meters: the flow on pipe 1–2 at 1.10, the flow on pipe 2–3 at -0.20, and the demand at junction 2 itself, from a meter on the zone it supplies, at -1.50, all in kg/s. The balance at junction 2 says that what arrives on the two pipes is what the zone draws, $m_{12} - m_{23} = -q_2$; the meters say 1.30 on the left and 1.50 on the right. No state of the exact model satisfies all three to their accuracies, and the tight model of the first row of Table 4.5 answers by keeping its laws and spoiling its meters: the pipe flows are fitted at

Table 4.5: Inconsistent meters on the water loop, $m_{12} = 1.10$, $m_{23} = -0.20$, $q_2 = -1.50$ kg/s, each to 0.02: posterior as the width of the junction-2 balance is loosened while every other row stays at 0.001. The unmodeled leak is the violation of that balance at the posterior mean. About 3 is expected of an adequate model.

σ_{b_2}	unmodeled load	q_2	q_3	m_{12}, m_{23} fitted	misfit
0.001	0.000 ± 0.001	-1.434 ± 0.016	-0.631 ± 0.043	1.166, −0.268	33.5
0.01	0.015 ± 0.010	-1.439 ± 0.017	-0.636 ± 0.043	1.161, −0.263	31.0
0.1	0.178 ± 0.033	-1.493 ± 0.020	-0.689 ± 0.044	1.106, −0.209	4.0
1	0.199 ± 0.035	-1.500 ± 0.020	-0.696 ± 0.044	1.099, −0.202	0.5

Table 4.6: The misfit at the posterior against its expectation under Proposition 4.2, for the cases of this chapter. The expectation is the number of factors minus the number of unknowns; the spread of an adequate model's misfit is the square root of twice that.

case	factors, unknowns	expected	misfit
chain, consistent readings, tight physics	8, 6	2 ± 2	0.07
chain, inconsistent readings, tight physics	8, 6	2 ± 2	74
chain, inconsistent, node-2 balance at 100 W	8, 6	2 ± 2	10.6
loop, inconsistent meters, tight physics	10, 7	3 ± 2.4	33.5
loop, inconsistent, junction-2 balance at 1 kg/s	10, 7	3 ± 2.4	0.5

1.166 and −0.268, three accuracies from their readings, the demand at −1.434, three from its meter, and the misfit is 34 where an adequate model with ten factors and seven unknowns would give about three. Loosen the balance at junction 2 and the posterior finds the explanation: at a width of 0.1 the balance is violated by 0.18 ± 0.03 kg/s and the misfit is 4; at a width of one it is violated by 0.20 ± 0.04, every meter is fitted to within an accuracy, and the misfit is 0.5. The violation is a draw of 0.2 kg/s at junction 2 that the graph does not show — a leak on a main, a standpipe, a connection never recorded — and the model has measured it from three meters without being told it exists. That is two tenths of a kilogram a second on a district drawing two, ten percent of the throughput, which is squarely inside the range a real distribution network loses and does not account for; the whole of Part VI's first half is this paragraph at scale. The right panel of Figure 4.4 draws the sweep beside the chain's.

Table 4.6 sets every misfit of the chapter beside its expectation. The consistent chain and the loosened loop sit below expectation, which says their widths are, if anything, generous; the two tight inconsistent models sit thirty and forty standard deviations above it, which is not a subtle signal; and the loosened chain, at 10.6 against 2 ± 2, sits four deviations above, which is the residual doubt the text described.

Two differences from the chain are instructive. The loop's leak is found at a width of 0.1 kg/s, a tenth of the flows, whereas the chain's needed ten to a hundred kg/s, a tenth to a whole flow; the reason is that the loop's three meters pin the violation from both sides, while the chain's two gauges see the leak only through the source prior. And the loop's misfit falls all the way to 0.5, below its expected value, while the chain's stops at 10.6; the loop's data are explained entirely by one unmodeled leak, and the chain's are not, which is the signal that the chain has a second thing wrong or a prior too narrow.

4.8 In practice

Scale the rows, then set relative widths. Divide every physics row by the typical size of what it balances so that its residual is of order one, and state widths as fractions of that. Relative widths of 10^{-3} reproduce the hard model on every example in this chapter at a condition number a solver does not notice; 10^{-5} is past the edge on a model of any size. Without the scaling, a width that is small for a kilowatt balance is enormous for a kg/s one.

Sweep the width once. Run the model at three widths a decade apart and check that the posterior has stopped moving. If it has not, the widths are too loose and the physics is being bent to fit the data; if the condition number has grown past 10^{10}, they are too tight. The sweep is cheap and it is the only way to know where a given model sits.

Read the misfit before the estimates. The sum of squared standardized residuals over every factor, compared with the number of factors minus the number of unknowns, is the first number to report. A misfit near its expectation says the model, the widths and the data are consistent; a misfit far above it says they are not, and no estimate from that posterior should be believed until the reason is found.

Loosen the row you suspect, not every row. An inconsistency is explained by violating some row. Loosening all of them lets the posterior spread the violation wherever it is cheapest, which is usually wrong and always uninformative. Loosening one row asks a question, "is it this?", and the misfit answers it: the chain's data were explained by a leak at node 2 and not by anything at node 1.

Report the violation as a measurement. The residual of a loosened row at the posterior, with its posterior spread, is an estimate of the unmodeled flow, and it should be reported in the row's units with its error bar, exactly as a measured quantity would be.

Keep the hard limit as a check. When the soft model surprises, tighten the widths and compare with the input-space form. If the two disagree, the widths were doing work the modeler did not intend.

4.9 What is missing

The chapter has said what the widths mean and shown what they do on a six-variable model where every posterior was computed by inverting a six-by-six matrix. It has not said how the same computation is organized when the matrix is a million by a million and sparse, which is Chapter 6, nor which variables can be computed without the others, which is Chapter 5. And it has assumed that every row is linear, so that the soft joint is Gaussian and the manifold is a plane. A nonlinear closure makes the manifold curved, the soft joint non-Gaussian, and Proposition 4.1 local rather than global; a discrete availability variable makes the manifold a union of pieces, one per value. Chapter 7 takes those up.

Notes and further reading

The soft model is the quadratic penalty method for equality-constrained least squares, and §4.4 and §4.5 restate what Nocedal and Wright [65] say about it: the penalized problem converges to the constrained one as the penalty weight grows, and its conditioning degrades as the weight grows, which is why the weight is chosen finite and why augmented-Lagrangian methods exist. The book keeps the plain penalty form because its normal equations are the same sparse system as the deterministic solve (Proposition 6.2 in Chapter 6) and because a finite width is wanted for its own sake, to admit violation. The relation between condition number and lost digits is in Higham [34].

Treating physics as a soft constraint with a modeler-chosen width, and letting the size of the violation at the posterior carry diagnostic meaning, is the practice of power-system state estimation, where zero-injection junctions are handled either as hard constraints or as pseudo-measurements of very high weight [1, 64], and of process data reconciliation, where the residual of a balance is the signature of a leak or a bad meter [57]. The leak example of §4.6 is the data-reconciliation argument on the smallest possible flowsheet. The chi-squared test named in §4.5 is the standard adequacy test of both fields and of least-squares theory generally; its assumptions are stated there because they are often omitted.

The input-space form of §4.2, with the physics eliminated and the posterior written over the inputs alone, is how Bayesian inverse problems are usually posed [37, 88]: the forward map is $X(\mathbf{u})$, and its derivative is the sensitivity that adjoint methods compute. The observation that this form is dense while the residual form is sparse, and that the choice between them is the choice between Chapter 14's gradients and Chapter 8's elimination, is the book's framing.

Summary

- With zero widths the joint lives on the constraint manifold $\mathbf{x} = X(\mathbf{u})$, of dimension equal to the number of inputs. Its natural form is the prior on the inputs carried along by the solve.
- Conditioning on the manifold gives the input-space posterior $p(\mathbf{u})\,\ell(X(\mathbf{u}))$: exact, dense, and one solve per evaluation. The worked example of Chapter 3 was this.
- With finite widths the joint is a density on the full space, concentrated within about a width of the manifold. As the widths go to zero it converges to the hard posterior; the soft model is the quadratic penalty form of the constrained one.
- Soften because it keeps the problem unconstrained, keeps the graph sparse, and makes a violated law visible and measurable.
- Widths come from instruments, data sheets, forecasts, and known model error; balance widths are small relative to their flows, and the condition number grows as the inverse square of the relative width.
- On the chain, widths of 0.1 to 0.01 kg/s reproduce the hard posterior exactly; inconsistent readings through tight physics produce an absurd state with a misfit of 74; loosening one balance turns the inconsistency into a 58 ± 7 kg/s leak estimate and a misfit of 10.6.
- On the water loop the same sweep converges at a relative width of 10^{-3}, and three inconsistent meters become an unmodeled load of 0.20 ± 0.04 kg/s at junction 2 with the misfit falling from 34 to 0.5. Loosening the wrong row explains nothing, and the misfit says so.

Exercises

Exercise 4.1. For the linear physics $\mathbf{F} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$ with $\mathbf{A}$ square and invertible, write $X(\mathbf{u})$ explicitly, verify that $\partial X/\partial \mathbf{u} = -\mathbf{A}^{-1}\mathbf{B}$, and show that this matrix is dense for the pumping chain even though $\mathbf{A}$ and $\mathbf{B}$ are sparse.

Exercise 4.2. Prove item 3 of Proposition 4.1: in a Gaussian posterior whose precision includes the term $\mathbf{h}\mathbf{h}^{\mathsf{T}}/\sigma^2$ for a linear residual $r = \mathbf{h}^{\mathsf{T}}\mathbf{z}$, the posterior variance of r is at most σ^2. Use the fact that adding a positive semidefinite matrix to a precision cannot increase any variance.

Exercise 4.3. Re-run the sweep of Table 4.2 with the two closure rows at width 1 kg/s and only the two balance rows swept. Explain, in terms of Figure 4.1, why the estimate of G converges to a different limit than in the table, and compute that limit by treating the closure widths as model error in the input-space form.

Exercise 4.4. In the leak example, loosen the balance at node 1 instead of node 2. Show that the posterior finds a different explanation of the same readings, state what physical defect it corresponds to, and compare the two misfits. Which explanation should the modeler prefer, and can the data alone decide?

Exercise 4.5. Derive the volume factor of Remark 4.1 for one solved variable and one input, $r(x, u) = 0$, by substituting r for x in the integral $\int \delta(r(x, u))\,\mathrm{d}x$. Then evaluate it for the closure $Q - k(h_1 - h_2)$ solved for h_1 with k as the input, and state when it is constant.

Solutions to selected exercises

Exercise 4.2. Write the posterior precision as $\mathbf{\Lambda} = \mathbf{h}\mathbf{h}^{\mathsf{T}}/\sigma^2 + \mathbf{M}$ with $\mathbf{M}$ positive semidefinite, the sum of every other factor's contribution. The posterior variance of $r = \mathbf{h}^{\mathsf{T}}\mathbf{z}$ is $\mathbf{h}^{\mathsf{T}}\mathbf{\Lambda}^{-1}\mathbf{h}$. For any positive definite $\mathbf{\Lambda}_1$ and positive semidefinite $\mathbf{M}$, $\mathbf{\Lambda}_1 + \mathbf{M} \succeq \mathbf{\Lambda}_1$ implies $(\mathbf{\Lambda}_1 + \mathbf{M})^{-1} \preceq \mathbf{\Lambda}_1^{-1}$, so adding $\mathbf{M}$ to a precision cannot increase any variance. But $\mathbf{h}\mathbf{h}^{\mathsf{T}}/\sigma^2$ alone is singular, so take $\mathbf{\Lambda}_1 = \mathbf{h}\mathbf{h}^{\mathsf{T}}/\sigma^2 + \epsilon\mathbf{I}$ with $\epsilon > 0$ small and $\mathbf{M} - \epsilon\mathbf{I} \succeq 0$ for ϵ below the smallest eigenvalue of $\mathbf{M}$ restricted to the directions where it is positive; then $\mathbf{h}^{\mathsf{T}}\mathbf{\Lambda}^{-1}\mathbf{h} \le \mathbf{h}^{\mathsf{T}}\mathbf{\Lambda}_1^{-1}\mathbf{h}$, and by the Sherman–Morrison identity $\mathbf{h}^{\mathsf{T}}\mathbf{\Lambda}_1^{-1}\mathbf{h} = \|\mathbf{h}\|^2/\epsilon \cdot (1 - \|\mathbf{h}\|^2/(\epsilon\sigma^2 + \|\mathbf{h}\|^2)) = \sigma^2\|\mathbf{h}\|^2/(\epsilon\sigma^2 + \|\mathbf{h}\|^2) < \sigma^2$. When $\mathbf{M}$ is singular in the direction $\mathbf{h}$ the bound is attained in the limit, which is why the posterior spread of every residual in Table 4.2 equals the width: the physics row is the only factor that constrains its own residual.

Exercise 4.4. Loosening the balance at node 1 to 100 kg/s, with every other row at 0.01, gives a violation of -46 ± 20 kg/s, a source of 102 ± 20, a room of -0.1 ± 1.7 metres, and a misfit of 68.5; at ten kg/s the violation is -10, and at one kg/s nothing has moved from the tight solution. The violation's sign says that 46 kg/s more than the source is entering the discharge's flow path: a second, hidden source at node 1. But the posterior has not been helped: the trunk main is still twenty metres below its forecast, the sensors are still off by three or four accuracies, and the misfit has fallen from 74 only to 68. The reason is in the physics. The two readings fix $h_1 - h_2 = 31$ metres and hence $m_{12} = 155$ kg/s whatever balance is loosened; a leak at node 2 lets m_{2a} be less than m_{12}, so that $h_2 - h_a$ can be 52 metres instead of 77 and the trunk main can return to its forecast, while a defect at node 1 changes only where m_{12} came from and leaves $m_{2a} = m_{12} = 155$, so the trunk main must still sit at -5. The data decide: the

node-2 explanation reaches a misfit of 10.6 and the node-1 explanation does not leave 68, and the modeler should prefer the first by a margin of nearly sixty units of misfit, which Chapter 12 will turn into a likelihood ratio. What the data cannot decide is whether the node-2 leak is the whole story, since 10.6 is still above the expected 2; that residual doubt is about the source prior, and only a meter on the source or a better data sheet can remove it.

Chapter 5

Factorization, separators, and treewidth

Chapter 3 read the factor graph informally: a measurement travels along every path, and two inputs that share an observed effect become correlated. This chapter makes the reading exact. It states what a factorization does and does not say about independence, gives the rule for reading conditional independence off the graph, shows that the ports of Chapter 1 are exactly the sets that rule singles out, and then turns to cost: what it takes to compute a marginal on a factor graph, why the answer is governed by a number called treewidth, and why a physical cut with few ports is a cheap place to divide a computation. Nothing in the chapter requires Gaussians; everything in it holds for any positive factorized density.

5.1 Factorization is not independence

Two variables are *independent* if their joint density is the product of their marginals, $p(x_1, x_3) = p(x_1)\, p(x_3)$; knowing one says nothing about the other. They are *conditionally independent given* a third if the same holds for every fixed value of the third:

$$p(x_1, x_3 \mid x_2) \;=\; p(x_1 \mid x_2)\, p(x_3 \mid x_2), \qquad \text{written } x_1 \perp x_3 \mid x_2. \tag{5.1}$$

Neither implies the other.

Take the simplest factor graph with three variables, the chain of Figure 5.1:

$$p(x_1, x_2, x_3) \;\propto\; \varphi_A(x_1, x_2)\, \varphi_B(x_2, x_3). \tag{5.2}$$

Are x_1 and x_3 independent? In general, no. Integrate out x_2:

$$p(x_1, x_3) \;\propto\; \int \varphi_A(x_1, x_2)\, \varphi_B(x_2, x_3)\, \mathrm{d}x_2, \tag{5.3}$$

which is a function of x_1 and x_3 together that does not split into a product unless the factors are very special. Information passes from x_1 through x_2 to x_3, and the chain of Chapter 3 showed it passing: a reading two nodes away moved the source.

Are they independent *given* x_2? Yes, always. Fix x_2. The right side of equation (5.2) is then a function of x_1 alone times a function of x_3 alone, and a joint density that splits that way is a product of its marginals. That is equation (5.1), with one line of proof.

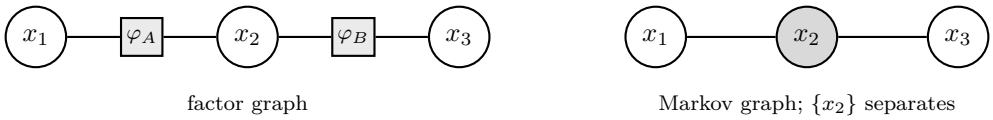


Figure 5.1: The three-variable chain. Left, as a factor graph. Right, its Markov graph: variables adjacent when they share a factor. Removing x_2 disconnects x_1 from x_3, so $x_1 \perp x_3 \mid x_2$. Nothing disconnects them otherwise, so in general $x_1 \not\perp x_3$.

So a factorization into local pieces does two opposite things at once. It creates global dependence, because every variable is reachable from every other along the factors. And it creates conditional independence, because fixing the variables on a path blocks it. The first is what makes a measurement anywhere informative everywhere. The second is what makes inference tractable, because it says which variables can be dealt with separately once a few others are known.

Principle 5.1. *Local factors create global dependence and, at the same time, conditional independence. Which variables are independent given which others is a property of the graph alone, and can be read before any number is computed.*

5.2 Reading independence off the graph

The reading uses the Markov graph of §2.5: one node per variable, an edge between two variables when some factor reads both.

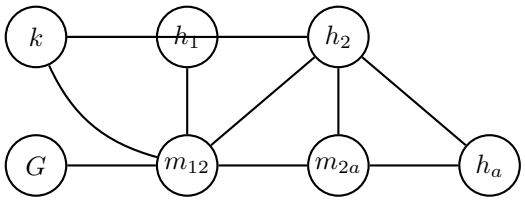
Definition 5.1 (Separator)**.** A set of variables S *separates* two disjoint sets A and B in the Markov graph if every path from a variable in A to a variable in B passes through some variable in S. Equivalently, removing the nodes of S leaves A and B in different connected components.

Proposition 5.1 (Separation implies conditional independence)**.** *Let $p(\mathbf{z}) \propto \prod_f \varphi_f(\mathbf{z}_f)$ be a factorized density and let S separate A from B in its Markov graph. Then $A \perp B \mid S$.*

Proof. Every factor's scope is a set of mutually adjacent variables in the Markov graph, by construction. A set of mutually adjacent variables cannot contain one variable from A and one from B, since those two would then be adjacent and the path of length one between them would avoid S. So every factor reads variables from $A \cup S$ only, or from $B \cup S$ only (or from S only, which is both). Group the factors: $p(\mathbf{z}) \propto f(\mathbf{z}_A, \mathbf{z}_S)\, g(\mathbf{z}_B, \mathbf{z}_S)$. For fixed $\mathbf{z}_S$ this is a product of a function of $\mathbf{z}_A$ and a function of $\mathbf{z}_B$, which is conditional independence. □

The proof is the chain argument of the previous section, with "the middle variable" replaced by "a set that every path must cross". The proposition is the useful direction of a two-way result; the other direction, that every conditional independence a positive density has is visible as a separation in some factorization of it, is the Hammersley–Clifford theorem and is not needed here. Two things about it matter for physical models.

First, the proposition asks nothing of the factors except that they exist. They may be Gaussian, discrete, nonlinear, or indicator functions. So the independence structure of a model is settled by its factor graph, which is settled by its physics, before any width, prior or sensor is chosen.



$\{m_{12}, m_{2a}\}$ separates G from h_a
$\{h_1, h_2, m_{12}\}$ separates k from h_a
no set of size one separates G from h_a

Figure 5.2: The Markov graph of the soft pumping chain with its three freed inputs (Figure 3.1), before any sensor is attached. Every edge is a pair read by one factor: the balances join the flows to each other and to the source, the closures join each flow to the potentials at its ends and, for c_{12}, to the conductance. The path from G to h_a carries no marginal dependence, and every path from either to the other crosses the two flows.

Second, the converse direction needs the density to be positive everywhere, and a model with hard physics is not: the delta factors of Chapter 4 put zero density off the manifold. Hard physics can therefore carry independencies that no separation shows. The soft model is positive, and the correspondence between graph and independence is then as clean as the proposition makes it. This is one more reason, after the three of §4.4, to compute on the soft graph.

5.3 Explaining away, precisely

Chapter 3 found that the source G and the room h_a, independent under the prior, were correlated at -0.985 after the reading of h_2. Proposition 5.1 says when two variables *are* independent given a set; it does not say when they are not, and it says nothing about marginal independence, which is independence given the empty set. Both are needed to make the observation exact.

Look at the Markov graph of Figure 3.1, drawn in Figure 5.2. The source is adjacent to m_{12} through the balance b_1; the room is adjacent to m_{2a} and h_2 through the closure c_{2a}; and m_{12}, m_{2a}, h_2 are all adjacent to one another through b_2 and c_{12}. There is a path from G to h_a, so the empty set does not separate them, and the proposition is silent about their marginal independence.

Yet under the prior they are independent, by construction: their priors are separate unary factors, and Chapter 4 showed that for linear physics the physics factors integrate out to a constant, leaving the product of the priors. The Markov graph draws an edge path that carries no marginal dependence.

The reason is that the physics between them is, in the hard limit, a deterministic map from inputs to solved variables. Marginalizing a deterministic map's outputs returns its inputs untouched. The undirected graph cannot express "this path carries dependence only when something on it is observed", because it has no notion of input and output. The directed language of §2.2 can: the pattern $G \to h_2 \leftarrow h_a$, two arrows meeting head to head at a common effect, is called a *collider*, and the rule for directed graphs is that a path through a collider is blocked when the collider is *not* observed and opened when it is, or when anything downstream of it is. That is the opposite of the rule for a chain, and it is the whole of explaining away: two independent causes of a common effect become dependent once the effect is seen, because having seen the effect, the more one cause explains it the less the other needs to.

For the models of this book the statement can be made without the directed formalism, because the physics has the input-output structure of Chapter 4 built in.

Proposition 5.2 (Explaining away on a physical graph)**.** *Let the physics be linear with widths $\sigma > 0$, and let two inputs u_1, u_2 carry independent priors. Then $u_1 \perp u_2$ under the prior predictive; and if a solved variable x_j that depends on both, $\partial X_j/\partial u_1 \neq 0 \neq \partial X_j/\partial u_2$, is observed, then in general $u_1 \not\perp u_2 \mid y_j$.*

Proof. For the first half, write the physics as $\mathbf{F} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$ with $\mathbf{A}$ square and invertible, as in Chapter 4. The prior predictive of the inputs is the joint with the solved variables integrated out, $p(\mathbf{u}) \int \exp(-\|\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}\|^2/2\sigma^2)\,\mathrm{d}\mathbf{x}$. Substitute $\mathbf{x} = -\mathbf{A}^{-1}\mathbf{B}\mathbf{u} + \mathbf{A}^{-1}\mathbf{r}$; the Jacobian of the substitution is $1/|\det \mathbf{A}|$, a constant, and the integral becomes $|\det \mathbf{A}|^{-1} \int \exp(-\|\mathbf{r}\|^2/2\sigma^2)\,\mathrm{d}\mathbf{r}$, which does not depend on $\mathbf{u}$. So the prior predictive of the inputs is the prior, $p(u_1)\,p(u_2)$, a product, and $u_1 \perp u_2$. For the second half, condition the same joint on a reading of x_j; by equation (4.3) the posterior of the inputs is $p(u_1)\,p(u_2)\,\ell(X_j(u_1, u_2))$ in the hard limit, and for finite σ the same with ℓ smoothed as in the proof of Proposition 4.1. A product of a function of u_1, a function of u_2, and a function of both factorizes into a function of u_1 times a function of u_2 only if the third factor does, which for a Gaussian ℓ of a linear X_j with both derivatives nonzero it does not: $\ell(au_1 + bu_2)$ with $ab \neq 0$ contains the cross term $-abu_1u_2/\sigma_y^2$ in its exponent, and a joint Gaussian with a nonzero cross term is not a product. Hence $u_1 \not\perp u_2 \mid y_j$. □

The second half is read from the input-space posterior, and the cross term is the correlation of equation (3.11). In the worked example X_j was $h_a + 0.5\,G$, the likelihood was a Gaussian in that combination, and the posterior was a ridge along it. The ridge is the picture of explaining away: along the ridge, a higher source is paid for by a lower room.

> **Principle 5.2.** *Independent inputs stay independent while nothing downstream of them is observed, and become dependent the moment a shared consequence is. The undirected graph over-connects them; the input-output structure of the physics is what says which is which.*

5.4 Ports are separators

Now return to the boundary. Chapter 1 cut a physical graph into a set S of nodes and its complement, and found that the flows across the cut sum to the net source inside, whatever the inside does (Proposition 1.1). The probabilistic version of that statement is that the cut carries everything one side needs to know about the other.

Take a physical cut through a set of edges $\mathcal{C}$. For each cut edge e with inside end i and outside end j, the closure factor c_e reads the flow F_e, both potentials ϕ_i and ϕ_j, and the edge's parameters. Every other factor reads variables from one side only, together with the cut flows: the balance at i reads F_e and the inside flows at i; the balance at j reads F_e and the outside flows at j. Figure 5.3 draws the Markov graph near one cut edge.

Remove the pair $\{F_e, \phi_j\}$. The closure's remaining scope is $\{\phi_i\}$ and the edge's parameters, all inside; the outside balance at j has lost its only inside-facing variable. No path remains from inside to outside through edge e. Do the same for every cut edge and the two sides are disconnected.

Proposition 5.3 (Ports separate)**.** *For a physical cut $\mathcal{C}$, the set of port pairs $\{(F_e, \phi_{j(e)}) : e \in \mathcal{C}\}$, one flow and one potential per cut edge, separates the inside variables from the outside variables in the Markov graph of the physics. Hence, given the ports, the inside and the outside are conditionally independent.*

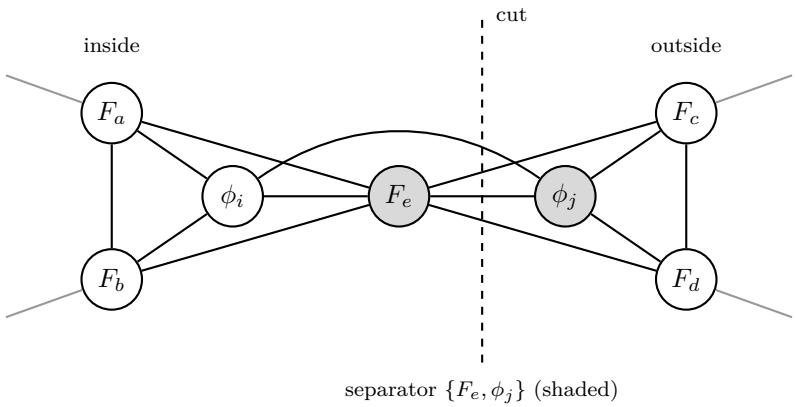


Figure 5.3: The Markov graph around one cut edge e from inside node i to outside node j. The closure c_e makes $\{F_e, \phi_i, \phi_j\}$ a clique; the balance at i joins F_e to the other inside flows F_a, F_b; the balance at j joins it to the outside flows F_c, F_d. Removing the shaded pair $\{F_e, \phi_j\}$ leaves no path from the inside cluster to the outside one. The pair is the port of the edge.

Two remarks. The separator is not unique; $\{F_e, \phi_i\}$ works as well, and so does $\{F_e, \phi_i, \phi_j\}$, and which potential is included is the choice of which side owns the cut's potential. But one flow and one potential per edge is the smallest set that works, and neither alone suffices: the potentials without the flow leave the balances at i and j joined through F_e, and the flow without a potential leaves the closure joining ϕ_i to ϕ_j. The pair that Chapter 1 called a port, on physical grounds, is the pair that separation requires on graphical grounds.

And the proposition is the probabilistic reading of Proposition 1.1. That proposition said the cut flows are exact: the inside's detail cancels, and the outside sees only what crosses. This one says the cut variables are *sufficient*: given them, the inside's detail is irrelevant to the outside, and vice versa. A region can be summarized to its neighbors by a function of its ports alone, with nothing lost. What that function is, and how it is computed, is the subject of Chapter 8.

Principle 5.3. *The ports of a physical cut are a separator of the factor graph. Given the ports, each side of the cut is conditionally independent of the other, so each side can be summarized to the other by a function of the ports alone.*

5.5 Elimination and fill

Independence says what can be computed separately. Cost says what it takes. The basic operation of exact inference is to remove one variable from the factor graph by integrating it out, and the cost of inference is the cost of doing that repeatedly in some order.

5.5.1 One elimination

Suppose x_1 appears in two factors, $\varphi_A(x_1, x_2)$ and $\varphi_C(x_1, x_3)$, and nowhere else. Integrating it out of the joint replaces the two factors by one,

$$g(x_2, x_3) \;=\; \int \varphi_A(x_1, x_2)\, \varphi_C(x_1, x_3)\, \mathrm{d}x_1, \tag{5.4}$$

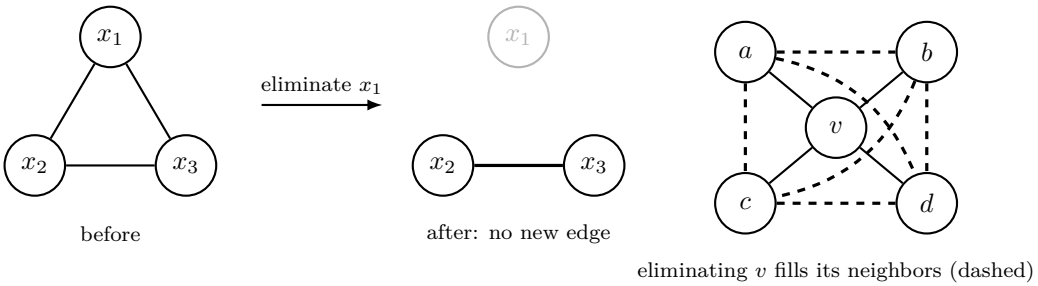


Figure 5.4: Elimination and fill. Left: eliminating x_1 from the triangle creates a factor on $\{x_2, x_3\}$, but they were already adjacent, so the graph gains no edge. Right: eliminating a variable with four neighbors that were not adjacent to each other creates six fill edges; the neighbors become a clique of four, and the factor on that clique is the largest object the computation must hold.

which reads every variable the eliminated factors read except x_1. Nothing else in the joint changes, since no other factor read x_1. The result is a factor graph on one fewer variable that has exactly the marginal of the original over the remaining ones.

The new factor is the point. Before elimination, x_2 and x_3 shared no factor; after it, they share g. In the Markov graph an edge has appeared between x_2 and x_3. Such an edge is called *fill*, and eliminating a variable always fills in every pair of its neighbors that was not already adjacent: the neighbors of the eliminated variable become a clique. Figure 5.4 shows it on a triangle and on a variable with four neighbors.

5.5.2 Order and width

Eliminate all the variables but one, in some order, and what remains is the marginal of that one. The cost is dominated by the largest factor created along the way. For discrete variables with K states, a factor on $w+1$ variables is a table of K^{w+1} numbers, and creating it costs that many operations. For Gaussian variables a factor on $w+1$ variables is a $(w+1) \times (w+1)$ block, and eliminating a variable from it costs of order w^2; the total for the whole graph is what a sparse Cholesky factorization costs, and the fill edges are the nonzeros of the factor that were not in the original matrix.

The largest factor depends on the order. Eliminate the middle of a chain first and the two ends become adjacent; eliminate from one end and no fill ever appears. For the chain the best order creates factors on two variables at most; for the four-neighbor star of Figure 5.4, eliminating the center first creates a factor on four, while eliminating the leaves first creates factors on two. The number that measures how well the best order can do is:

Definition 5.2 (Treewidth)**.** The *width* of an elimination order is one less than the size of the largest factor scope it creates. The *treewidth* of a graph is the minimum width over all elimination orders.

A tree has treewidth one: eliminate leaves inward and every factor created has two variables. A single cycle has treewidth two. A $\sqrt{n} \times \sqrt{n}$ grid has treewidth $\sqrt{n}$. A complete graph on n variables has treewidth $n-1$. Finding the optimal order is hard in general, but good orders are found in practice by heuristics that eliminate low-degree variables first, or that recursively cut the graph at small separators and eliminate the pieces before the separator.

Two bounds fix the number without a search.

Lemma 5.1 (Bounds on the width)**.** *The treewidth of a graph is at least the size of its largest clique minus one, and at most the width of any elimination order. A graph is a tree if and only if its treewidth is one.*

Proof. Consider a largest clique Q and the first of its vertices to be eliminated in any order. Every other vertex of Q is still present and adjacent to it, since elimination never removes an edge, so its front has at least $|Q| - 1$ vertices; the width of the order, and hence the minimum over orders, is at least $|Q| - 1$. The upper bound is the definition. For the last statement, a tree has an order of width one, leaves inward, and a graph with a cycle contains, after eliminating everything off the cycle, a cycle whose first eliminated vertex has two neighbors, so its width is at least two. □

The lower bound is often tight on physical graphs. The water loop's row-per-factor Markov graph has a clique of three, the scope of the pipe closure c_{23}, so its treewidth is at least two, and an order of width two exists (Exercise 5.3); every closure that reads a flow and two potentials plants such a triangle, which is why no row-per-factor physical graph is a tree.

> **Principle 5.4.** *Exact inference costs what the largest factor created by elimination costs, and that is governed by the treewidth of the factorization, not by the number of variables. Sparse is not enough; a sparse graph with large treewidth is expensive.*

5.5.3 What this means for a physical graph

Return to the two examples. The pumping chain in its row-per-factor form has a six-cycle (§2.1), so its treewidth is two; in nodal form, with the flows eliminated first, it is a path and its treewidth is one. Eliminating the flows is what the nodal form of §2.4 did by hand. A flow appears in one closure and two balances, and eliminating it joins everything those three factors read: the potentials at its ends and the other flows at both its nodes. That fill among flows is temporary. Once every flow is gone, what remains joins two potentials exactly when their nodes share an edge, which is the network's own graph, and no fill survives. That is why power engineers write the nodal equations first: the reduced conductance matrix has the sparsity of the network itself. Whether to eliminate flows first or last in a numerical factorization is a different question, answered by counting the fill, and Chapter 6 counts it.

A branched service zone is a tree, so in nodal form its treewidth is one and exact inference on it is linear in its size. A meshed transmission network has treewidth larger than one, but it is a sparse, nearly planar graph, and planar graphs have separators of size $O(\sqrt{n})$ [53], so their treewidth grows no faster than the square root of the number of junctions and in practice much more slowly; the measured values for the standard test networks are reported in Chapter 18. Exact inference on them is affordable. A finite-element discretization of a three-dimensional solid is harder: its separators, hence its treewidth, grow as $n^{2/3}$ [61], so sparse direct factorization becomes expensive at scale, which is one motivation for iterative and domain-decomposition solvers.

The physical cut of §5.4 is the elimination strategy the last paragraph hinted at. Cut the network at a set of ports; eliminate everything inside each region, which creates fill only within the region and among its ports; then what remains is a factor graph on the ports alone. The largest factor created inside a region is bounded by the region's own treewidth, and the factor left on the ports has scope equal to the number of ports. The cost of the whole is set by the

Table 5.1: Elimination orders on the water loop in row-per-factor form, five variables: the width distribution over all 120 orders, and the widths and fill of the orders that eliminate the flows first or the heads first.

orders	count	width	fill edges
all orders	120	2: 24 orders; 3: 72; 4: 24	–
flows first	12	2 to 4	0 to 3
heads first	12	3	2
m_{23} first	24	4	3 or more

region sizes and the port counts, and a cut with few ports is a cheap cut. Chapter 8 carries this out for Gaussian models, where it is the Schur complement, and Chapter 18 asks how to choose the cut.

5.6 Worked examples: counting width and fill

The claims of the last section can be counted on the book's examples, and the counts are in `ch05_separators.py`.

The water loop, every order. The water loop in row-per-factor form has five variables and seven Markov edges (Figure 2.6); Table 5.1 counts its orders. Of its 120 elimination orders, 24 have width two, 72 width three, and 24 width four; none has width one, because the Markov graph contains the triangle $h_2h_3m_{23}$ — a triangle of *variables*, not of pipes — and a graph with a triangle in it is not a tree. The twelve orders that eliminate the three flows first have widths from two to four depending on the order among the flows, and create between none and three fill edges, all among the flows themselves; what remains is the nodal form, two heads joined by the edge that c_{23} had already drawn. The twelve orders that eliminate the two heads first all have width three and create two fill edges, m_{12} to h_3 and then m_{12} to m_{13}, joining flows that no factor read together. Flows first is the better family, but not uniformly: within it, eliminating m_{23} first, the flow on the pipe that closes the loop, costs width four.

The two-district network. Table 5.2 counts the same things on the network of two loops and a transfer main that Chapters 8 and 15 use, drawn in Figure 5.5. Its nodal Markov graph, the five heads with the network's own edges, has treewidth two. Its row-per-factor graph, twelve variables and twenty-six edges, has treewidth at most four by the minimum-fill heuristic, which found an order of width four with three fill edges. Eliminating all seven flows first has width six and creates fourteen fill edges before the flows are gone, because a flow's elimination joins every other flow at both its junctions; eliminating the heads first has width eight and twenty-six fill edges. The lesson of §5.5 stands corrected in one respect: the nodal form is the right *result*, with the network's own sparsity, but reaching it by eliminating every flow before any head is not the cheapest *route*, and a heuristic that interleaves them is. The same table checks Proposition 5.3: removing the transfer flow m_{34} and the head h_4 disconnects district A from district B, and removing either alone, or the two heads at the transfer main's ends without the flow, does not.

Lemma 5.1 brackets these numbers. The two-district graph's largest clique is the scope of a pipe closure, three variables, so its treewidth is at least two, and the minimum-fill order shows it is at most four; the true value lies between, and finding it exactly is the hard problem the Notes

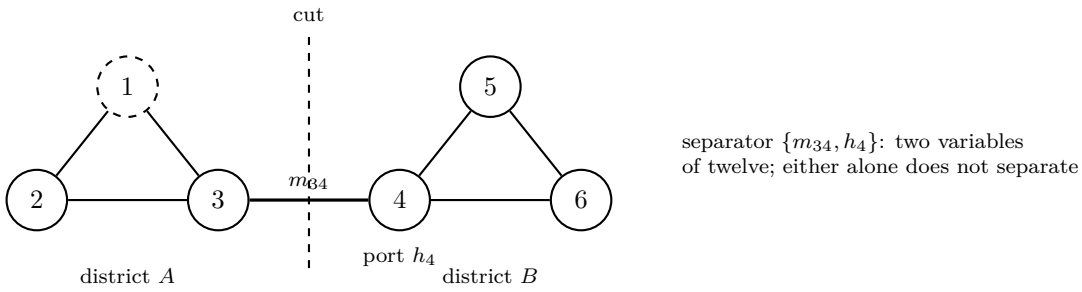


Figure 5.5: The two-district network cut at its transfer main. The port pair of the cut edge, the transfer flow and the head on the far side, separates the two loops in the row-per-factor Markov graph; the script confirms that neither variable alone does, nor the two end heads without the flow.

Table 5.2: Elimination on the two-district network in row-per-factor form (twelve variables): width and fill under three orders, and which variable sets separate the two loops.

order	width	fill edges
all flows first, then heads	6	14
all heads first, then flows	8	26
minimum fill (interleaved)	4	3
removed	**left and right loops**	
$\{m_{34}, h_4\}$	separated	
$\{m_{34}\}$ or $\{h_4\}$ alone	connected	
$\{h_3, h_4\}$	connected	

cite. On graphs of this size the gap is harmless, since a width of four is as cheap as a width of two; on a network of a thousand junctions the heuristics' orders are what a solver uses, and Chapter 18 reports what they find.

The cooling loop. The two-domain model of Example 1.3 has a Markov graph of twenty variables and forty-six edges with treewidth at most six by the heuristics and at least two by its largest clique, the scope of a carrier closure. Its interesting separator is not a physical cut but a domain cut: remove the six mass flows and the four pressures are disconnected from the four temperatures and six energy flows. Given the mass flows, the pressure layer and the temperature layer are conditionally independent, which is the probabilistic statement of what Chapter 2's Figure 2.4 showed structurally: the domains couple only through the flows the carrier closures read. A pressure measurement informs a temperature only by way of what it says about a mass flow. The minimum separator between one particular pressure and one particular temperature is smaller, two variables, the mass flow and energy flow of the return edge, which is the physical statement that the return header's temperature depends on the supply pressure only through what returns.

The ring, around and across. A ring of twelve junctions, each tied to its two neighbors, has treewidth two: eliminate the junctions in order around the ring and every elimination joins the two neighbors of the eliminated junction, a front of two, creating nine fill edges in all (the last

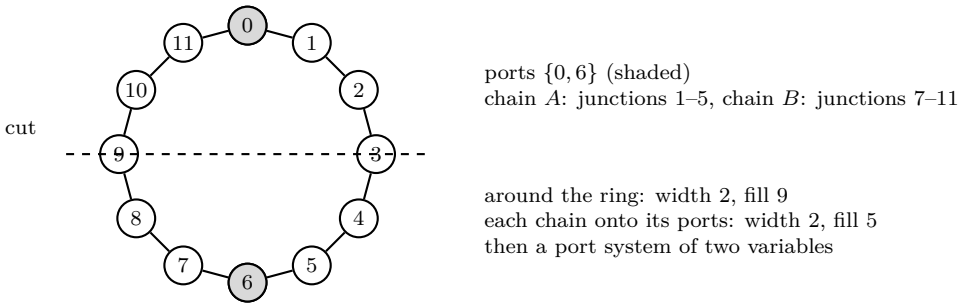


Figure 5.6: A ring of twelve junctions cut at two opposite junctions into two chains. Eliminating each chain's interior onto the shaded ports costs the same width as going around the ring, and leaves a two-variable port system that the chains share; the gain is independence and reuse, not width.

two eliminations find their neighbors already adjacent). Figure 5.6 shows the alternative. Cut the ring at two opposite junctions, which are then the ports of two chains of five junctions each; eliminate each chain's interior onto its ports, five fill edges and a front of two per chain; and what remains is a port system on two variables. Nothing has been gained in width, which was already two, but the two chains can be eliminated independently, each holds only its own factors, and the port system is what the chains need to exchange. This is the structure of Chapter 8's region messages in the smallest possible case, and the reason to prefer it is not the width but the reuse: a change in one chain recomputes that chain's five eliminations and the two-variable port solve, not the ring.

Grids against networks. Figure 5.7 plots the elimination width of square grids, the graph of a grid network discretized on $s \times s$ nodes, from 4×4 to 16×16, against the number of nodes, with $\sqrt{n}$ for comparison. The width grows as $\sqrt{n}$ and a little faster, 22 at 256 nodes, as the separator theorem for planar graphs says it must. The two-district network and the cooling loop sit far below the curve at their sizes. A grid is the hardest two-dimensional physics there is for elimination, because every node has four neighbors and no cut is thin; a network built by engineers has long thin regions joined at few ports, and Chapter 18 measures how far below the curve real networks of a thousand junctions sit.

5.7 Why separators matter for cost

The chapter closes with the count that motivates Part III.

Take a system that divides into three regions, each containing twenty binary uncertain quantities (component availabilities, say), and suppose the physics makes the regions conditionally independent given a separator S of k variables between them. The joint has 2^{60} states, about 10^{18}. A method that must visit each state cannot be run.

Proposition 5.1 says the joint factorizes across the separator,

$$p(R_A, R_B, R_C, S) \;\propto\; f_A(R_A, S)\, f_B(R_B, S)\, f_C(R_C, S), \tag{5.5}$$

for suitable factors, where R_A denotes region A's variables. Then any marginal of interest can

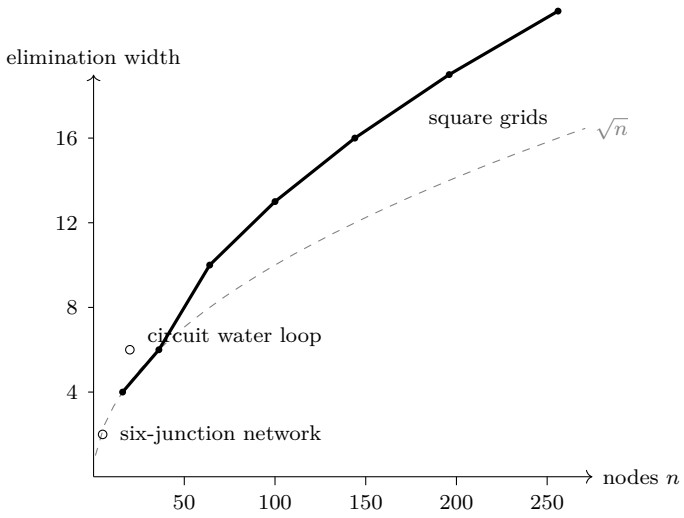


Figure 5.7: Elimination width of square grids of n nodes (filled points, minimum-fill ordering) against $\sqrt{n}$ (dashed). The six-junction network and the cooling loop are marked at their sizes. A grid's width grows with the square root of its size; the networks' widths are set by their ports.

Table 5.3: Operations to marginalize three regions of n binary variables exactly: joined at one separator of k variables (star), or in a chain with two separators of k each, against enumeration of the joint.

n	k	star	chain	ratio	enumeration 2^{3n}
20	5	1.0×10^{8}	1.1×10^{9}	11	1.2×10^{18}
20	3	2.5×10^{7}	8.4×10^{7}	3.3	1.2×10^{18}
30	5	1.0×10^{11}	1.2×10^{12}	11	1.2×10^{27}

be found by eliminating each region into a message on the separator,

$$m_A(S) = \sum_{R_A} f_A(R_A, S), \quad m_B(S) = \sum_{R_B} f_B(R_B, S), \quad m_C(S) = \sum_{R_C} f_C(R_C, S), \tag{5.6}$$

and combining them, $p(S) \propto m_A(S)\, m_B(S)\, m_C(S)$. Each message costs $2^{20} \cdot 2^k$ operations at most, about $10^6 \cdot 2^k$, and the combination costs 2^k. For a separator of five variables the whole computation is a few times 10^7 operations, against the 10^{18} of enumeration. The exponent that matters is k, the size of the separator, not 60, the number of uncertain quantities.

Table 5.3 tabulates the count for three sizes, beside the chain arrangement of Exercise 5.5 in which the middle region touches two separators and pays for both; Figure 5.8 draws the two arrangements.

This is the central fact about factorized inference, and it is worth stating what it does and does not claim. It claims that the cost of *exact* inference scales with the separator, and that a physical system with a natural cut of few ports has a small separator for free. It does not claim that a method which samples states instead of enumerating them is defeated by 2^{60}; a sampler draws a hundred thousand states from a space of 10^{18} without difficulty, and the count above is not an argument against it. The argument that does bear on samplers is about what each state costs to evaluate, and about how much of that evaluation the messages (5.6) let one reuse across

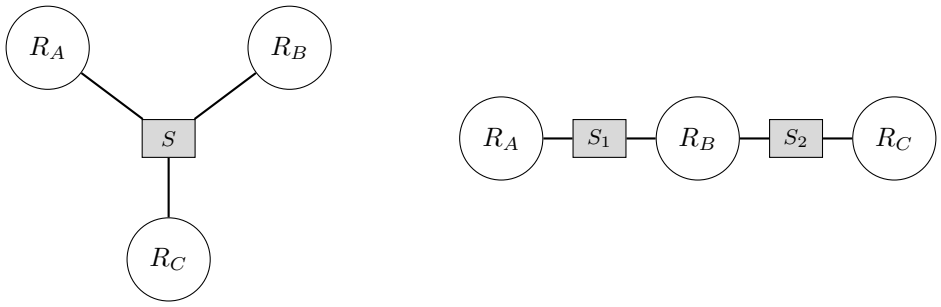


Figure 5.8: Three regions joined at one separator (left) and in a chain through two (right). In the chain the middle region's message must be computed jointly over both separators it touches, and its cost is exponential in their total size.

related questions. Chapter 10 makes that argument with numbers.

5.8 In practice

Read independence before computing. The Markov graph is free, and Proposition 5.1 answers, without a number, which quantities a proposed sensor can and cannot inform. If every path from the sensor to the quantity of interest crosses a set of variables that are already known, the sensor is wasted.

Remember which paths are open. Two independent inputs of a physical model are correlated only after something downstream of both is observed. A report that lists the sources as independent after the readings are in has misread the graph.

Cut at the ports. A physical cut's port pairs are the minimal separator; the potentials alone or the flows alone are not. A region summarized on anything less than its ports leaks information, and on anything more wastes it.

Do not trust "flows first". The nodal form is the right final graph; the cheapest route to it interleaves flows and potentials. Let a fill-reducing heuristic choose the order, and check the width it reports against the network's own sparsity.

Measure the width of your graph. A minimum-degree ordering gives it in milliseconds. A physical network of any size has a small width; a discretized continuum does not; and a model whose width is unexpectedly large usually contains a closure that reads more variables than the physics requires.

Count the separator, not the state space. For an exact computation the cost is exponential in the separator's size and linear in the number of regions. Before declaring a problem intractable at 2^{60} states, find the cut, count its ports, and count again.

5.9 What is missing

The chapter has read independence off the graph and counted the cost of elimination without saying what a factor is or how one is integrated out. For the Gaussian model both are matrices, elimination is Gaussian elimination, fill is fill, and the treewidth argument becomes a statement about sparse Cholesky. That is Chapter 6. Chapter 7 asks what elimination means when a factor is not Gaussian, and Chapter 8 organizes elimination by regions and ports, as §5.4 promised.

Notes and further reading

Markov properties of undirected graphs, the global Markov property that Proposition 5.1 states, the positivity condition and the Hammersley–Clifford theorem that supplies the converse are treated in Lauritzen [49] and Koller and Friedman [45]; the theorem's first published proof is Besag's [7]. Explaining away, colliders and the d-separation rule that §5.3 paraphrases are Pearl's [70]. Proposition 5.2 is the book's specialization of those rules to a physical graph whose input-output structure is supplied by the physics rather than by declared arrows.

Vertex elimination, fill, and the correspondence between elimination orders and chordal completions are due to Rose, Tarjan and Lueker [78]; that minimizing fill is NP-complete is Yannakakis's result [102], and that deciding treewidth is NP-complete is Arnborg, Corneil and Proskurowski's [3]. Elimination as the basis of exact inference on graphical models, and the junction tree that Chapter 8 uses, are from Lauritzen and Spiegelhalter [50]. Nested dissection, the strategy of §5.5's last paragraph, is George's [25]; the separator theorems that bound treewidth for planar and finite-element graphs are Lipton and Tarjan's [53] and Miller, Teng, Thurston and Vavasis's [61]. Davis [16] is the reference for all of this from the sparse-matrix side.

Proposition 5.3 is the book's. Its two halves are inherited: the port as a conjugate flow-potential pair is the bond-graph notion [39, 69, 95], and separation as sufficiency is the graphical-model notion. Their combination, that the physical port is the minimal probabilistic separator of the physics, is the construction on which Part III and Part V rest. The three-region count of §5.7 is the standard argument for the sum-product algorithm [47]; the caveat about samplers is the book's, and Chapter 10 is its development.

Proposition 5.3 has a concrete consequence in water networks. Han, Eguchi, Mehrotra and Venkatasubramanian [33] partition a network at its articulation junctions and estimate the biconnected components separately. An articulation junction's head is a separator of the network graph, but by itself it is not a separator of the factor graph: the balance factor at that junction contains the flows of every incident pipe, so the two sides stay coupled through the flow of the bridge until that flow joins the head in the separator, which is the head-flow port of §5.4. On EPANET's Net3 example, eliminating the 31 articulation heads alone leaves one connected interior block; eliminating the 31 heads and the 32 bridge flows splits it into 18. The elimination orderings and fill heuristics of §5.5 are used in the same way, and with the same sparse-matrix vocabulary, by Dellaert and Kaess [17], whose Bayes tree is the junction tree of an elimination order read as a directed structure.

Summary

- A factorization creates global dependence and conditional independence together. Fixing a separator blocks every path.

- Separation in the Markov graph implies conditional independence, for any factors. Hard physics breaks the converse; soft physics restores it.
- Independent inputs stay independent while nothing downstream is observed and become dependent once a shared consequence is. The undirected graph over-connects; the physics' input-output structure decides.
- The port pairs of a physical cut, one flow and one potential per cut edge, are a separator: the probabilistic form of exactness at a cut.
- Elimination fills the neighbors of the eliminated variable. Cost is set by the largest factor created, hence by treewidth, hence by the factorization and the order. Nodal form is the right result; an interleaved order is the cheap route to it: on the two-district network, width four and three fill edges against width six and fourteen for flows-first.
- A grid's width grows as $\sqrt{n}$; a physical network's is set by its ports. The cooling loop's two domains are separated by its six mass flows.
- Three regions of twenty binary variables joined at a five-variable separator cost a few times 10^7 operations to marginalize exactly, not 10^{18}. The exponent is the separator, not the state count.

Exercises

Exercise 5.1. Draw the Markov graph of the soft chain of Figure 3.1 with all six variables and the sensor on h_2. Find the smallest separator between G and h_a, and the smallest between k and h_a. Verify by Proposition 5.1 that $G \perp h_a \mid \{m_{12}, m_{2a}\}$, and explain physically why knowing both flows makes the source and the room irrelevant to each other.

Exercise 5.2. Prove the first half of Proposition 5.2: for $\mathbf{F} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$ with $\mathbf{A}$ square and invertible, show $\int \exp(-\|\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}\|^2/2\sigma^2)\,\mathrm{d}\mathbf{x}$ does not depend on $\mathbf{u}$. Then give an example of a nonlinear physics for which it does, and say what that means for the prior predictive of the inputs.

Exercise 5.3. For the water loop in row-per-factor form (Figure 2.3), find an elimination order of width two and prove no order of width one exists. Then show that eliminating the three flows first produces the nodal form of Figure 2.5, and find the order among the flows that creates no fill and the one that creates the most.

Exercise 5.4. A ring of n junctions, each connected to its two neighbors, has treewidth two. Give an elimination order that achieves it and the fill it creates. Now cut the ring at two opposite junctions into two chains and eliminate each chain onto its ports. What is the largest factor created, and how does it compare with eliminating the ring in the naive order around it?

Exercise 5.5. Repeat the count of §5.7 for regions of n binary variables each joined at a separator of k, and for regions joined in a chain $A - S_1 - B - S_2 - C$ rather than at a single separator. Which arrangement is cheaper, and by what factor, when $k = 5$ and $n = 20$?

Solutions to selected exercises

Exercise 5.3. An order of width two: m_{12}, m_{13}, m_{23}, h_2, h_3. Eliminating m_{12} joins its neighbors h_2 (from c_{12}) and m_{23} (from b_2); they were not adjacent, so one fill edge appears and the front had two variables. Eliminating m_{13} likewise joins h_3 to m_{23}, a second fill edge, front of

two. Eliminating m_{23} now has neighbors h_2 and h_3, already adjacent through c_{23}: front of two, no fill. The two heads remain, joined, and the order's width is two. No order of width one exists because the Markov graph contains the triangle h_2, h_3, m_{23}: whichever of its three vertices is eliminated first has the other two as neighbors, a front of at least two. Among flows-first orders the script finds widths two, three and four: eliminating m_{23} first is worst, since its four neighbors h_2, h_3, m_{12}, m_{13} become a clique with three fill edges and a front of four; eliminating it last, after m_{12} and m_{13} have each been joined to it, is the order above, and its two fill edges are the ones that the nodal form needs and no more. The claim in §5.5 that flows-first creates no lasting fill is right about what survives, the head-to-head edges of the network, and silent about the temporary fill among flows, which the order among the flows controls.

Exercise 5.5. With three regions of n binary variables meeting at one separator of k, each message costs $2^n \cdot 2^k$ and the combination 2^k: about $3 \cdot 2^{n+k}$. For $n = 20$, $k = 5$ that is 1.0×10^8 operations. In the chain $A - S_1 - B - S_2 - C$, the end regions send messages on one separator each, 2^{n+k}, but the middle region's message must carry both separators, since B's factor reads S_1 and S_2 together: $2^n \cdot 2^{2k}$. The total is $2^{n+k} \cdot 2 + 2^{n+2k} + 2^{2k}$, about 1.1×10^9 for the same n and k: the chain costs eleven times more, and the factor is $2^k/3$ in general, because the middle region pays for two separators at once. Both are negligible beside enumeration's $2^{60} = 1.2 \times 10^{18}$. The lesson for a partition is that what a region pays for is the total size of the separators it touches, so a region in the middle of a chain is charged twice, and a star of regions around one separator is cheaper than a chain of the same regions.

Chapter 6

The linear-Gaussian case

Everything so far has been about structure: which variables share a factor, which are independent given which, what elimination costs. This chapter is about numbers. When every row is linear and every factor is Gaussian, the posterior is Gaussian, and a Gaussian is two objects: a matrix and a vector. The factor graph becomes the sparsity pattern of the matrix. Elimination becomes Gaussian elimination. Fill becomes fill. The most probable state is the solution of one sparse linear system, and the uncertainty of every variable is read from the same factorization that produced it. When a row is mildly nonlinear the same machinery runs a few times instead of once and returns a Gaussian approximation whose quality can be measured. The chapter ends with the pumping chain in matrix form, the fill of three elimination orders counted, and a nonlinear closure checked against exact quadrature.

6.1 The Gaussian in information form

A Gaussian density on $\mathbf{z} \in \mathbb{R}^n$ is usually written by its mean and covariance, $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. For a factor graph the other parameterization is the natural one:

$$p(\mathbf{z}) \;\propto\; \exp\Big(-\tfrac{1}{2}\,\mathbf{z}^{\mathsf{T}}\boldsymbol{\Lambda}\,\mathbf{z} + \boldsymbol{\eta}^{\mathsf{T}}\mathbf{z}\Big), \qquad \boldsymbol{\Lambda} = \boldsymbol{\Sigma}^{-1}, \quad \boldsymbol{\eta} = \boldsymbol{\Lambda}\boldsymbol{\mu}. \tag{6.1}$$

$\boldsymbol{\Lambda}$ is the *precision* or *information matrix* and $\boldsymbol{\eta}$ the *information vector*. The reason to prefer them is one line: the product of two Gaussians in information form is the Gaussian whose precision is the sum of the precisions and whose information vector is the sum of the information vectors. Multiplying factors is adding matrices.

Every factor of Chapter 3 is a Gaussian in a linear function of $\mathbf{z}$. Write a generic linear row as $r(\mathbf{z}) = \mathbf{h}^{\mathsf{T}}\mathbf{z} - c$ with width σ and weight $w = 1/\sigma^2$. Its factor $\exp(-w\,r^2/2)$ expands to

$$\exp\Big(-\tfrac{1}{2}\,w\,\mathbf{z}^{\mathsf{T}}\mathbf{h}\mathbf{h}^{\mathsf{T}}\mathbf{z} + w\,c\,\mathbf{h}^{\mathsf{T}}\mathbf{z}\Big) \times \text{const}, \tag{6.2}$$

which is equation (6.1) with $\boldsymbol{\Lambda} = w\,\mathbf{h}\mathbf{h}^{\mathsf{T}}$ and $\boldsymbol{\eta} = w\,c\,\mathbf{h}$: a rank-one matrix on the row's scope and a vector supported there. A prior on z_j is the case $\mathbf{h} = \mathbf{e}_j$, $c = \mu_j$: it adds w_j to one diagonal entry and $w_j\mu_j$ to one component. A sensor on z_o is the same with $c = y$. A physics row is $\mathbf{h}$ equal to the row's gradient.

Sum over all factors. Stack every residual, physics, prior and sensor, into one vector $\mathbf{g}(\mathbf{z})$ of length m_g, let $\mathbf{J}_g = \partial\mathbf{g}/\partial\mathbf{z}$ be its Jacobian and $\boldsymbol{\Omega} = \mathrm{diag}(w)$ its weights. Then

$$\boxed{\boldsymbol{\Lambda} \;=\; \mathbf{J}_g^{\mathsf{T}}\,\boldsymbol{\Omega}\,\mathbf{J}_g} \tag{6.3}$$

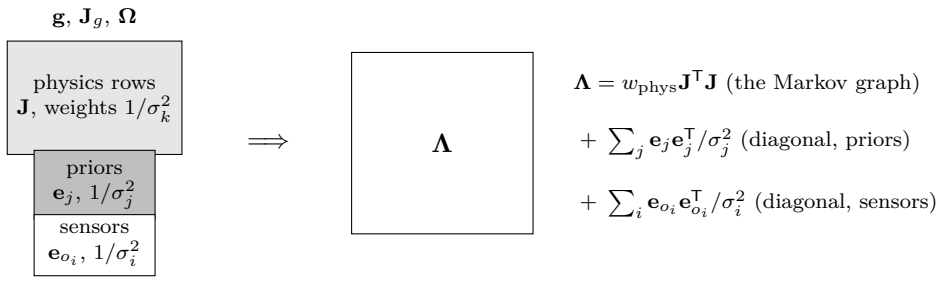


Figure 6.1: Assembling the precision. The three blocks of the stacked residual contribute three terms to $\mathbf{\Lambda}$: the physics rows give the sparse matrix whose pattern is the Markov graph, and priors and sensors add to the diagonal only. Removing the last two blocks and letting the physics weights grow recovers the deterministic solve (Proposition 6.2).

is the precision of the posterior, and the physics part of it is $w_{\text{phys}}\mathbf{J}^{\mathsf{T}}\mathbf{J}$ with $\mathbf{J}$ the Jacobian of Chapter 1, exactly the matrix that §2.5 said would appear. Figure 6.1 draws the assembly: the stacked residual has a block of physics rows, a block of priors and a block of sensors, and each block adds its own rank-m term to the same matrix.

Proposition 6.1 (Sparsity of the precision)**.** $\mathbf{\Lambda}_{ij} \neq 0$ *only if some row of* $\mathbf{g}$ *reads both* z_i *and* z_j*. The off-diagonal sparsity pattern of* $\mathbf{\Lambda}$ *is the Markov graph of the factor graph.*

The proof is that $(\mathbf{J}_g^{\mathsf{T}}\mathbf{\Omega}\mathbf{J}_g)_{ij} = \sum_k w_k (\mathbf{J}_g)_{ki}(\mathbf{J}_g)_{kj}$, and a term is nonzero only when row k has nonzeros in both columns. Two consequences follow at once. The number of nonzeros of $\mathbf{\Lambda}$ is linear in the size of the model, by the sparsity argument of §1.5. And every structural statement of Chapter 5, separation, fill, treewidth, applies to $\mathbf{\Lambda}$ verbatim, because the Markov graph is $\mathbf{\Lambda}$'s graph.

6.2 The most probable state

The negative log posterior, equation (3.8), in the stacked notation is

$$C(\mathbf{z}) \;=\; \tfrac{1}{2}\,\mathbf{g}(\mathbf{z})^{\mathsf{T}}\,\mathbf{\Omega}\,\mathbf{g}(\mathbf{z}), \tag{6.4}$$

and the most probable state, the *maximum a posteriori* or MAP estimate $\mathbf{z}^\star$, minimizes it. When $\mathbf{g}$ is linear, $\mathbf{g}(\mathbf{z}) = \mathbf{J}_g\mathbf{z} - \mathbf{c}$, the minimizer solves the normal equations

$$\mathbf{\Lambda}\,\mathbf{z}^\star \;=\; \mathbf{J}_g^{\mathsf{T}}\mathbf{\Omega}\,\mathbf{c} \;=\; \boldsymbol{\eta}, \tag{6.5}$$

one sparse symmetric positive-definite system. This is weighted least squares, and $\mathbf{z}^\star$ is also the posterior mean, since for a Gaussian the mode and the mean coincide.

When some rows are nonlinear, linearize at the current iterate, $\mathbf{g}(\mathbf{z}+\boldsymbol{\delta}) \approx \mathbf{g}(\mathbf{z}) + \mathbf{J}_g(\mathbf{z})\boldsymbol{\delta}$, and minimize the resulting quadratic:

$$\mathbf{\Lambda}(\mathbf{z})\,\boldsymbol{\delta} \;=\; -\,\mathbf{J}_g(\mathbf{z})^{\mathsf{T}}\mathbf{\Omega}\,\mathbf{g}(\mathbf{z}), \qquad \mathbf{z} \leftarrow \mathbf{z} + t\,\boldsymbol{\delta}. \tag{6.6}$$

This is the Gauss–Newton iteration. Because $\mathbf{\Lambda}$ is positive definite at a well-posed problem, $\boldsymbol{\delta}$ is a descent direction for C, and a step length $t \leq 1$ chosen by backtracking until C decreases makes the iteration converge from a poor start. Near the solution $t = 1$ and convergence is quadratic when the residuals are small, which for a physical model with tight physics they are.

Proposition 6.2 (Determinism is a corner of inference)**.** *Let the model carry no prior and no sensor, let the physics be square ($m = n$) with $\mathbf{J}$ nonsingular, and let all physics weights be equal. Then the Gauss–Newton step* (6.6) *is the Newton step for $\mathbf{F}(\mathbf{z}) = \mathbf{0}$, and any stationary point of C with $C = 0$ is a solution of the physics.*

Proof. With no other factors, $\mathbf{g} = \sqrt{w}\,\mathbf{F}$ and $\mathbf{J}_g = \sqrt{w}\,\mathbf{J}$. Equation (6.6) reads $w\mathbf{J}^\mathsf{T}\mathbf{J}\boldsymbol{\delta} = -w\mathbf{J}^\mathsf{T}\mathbf{F}$; cancel w and multiply by $\mathbf{J}^{-\mathsf{T}}$ to get $\mathbf{J}\boldsymbol{\delta} = -\mathbf{F}$, which is Newton's step. And $C = \frac{1}{2}w\|\mathbf{F}\|^2$ vanishes only where $\mathbf{F}$ does. □

The proposition is the precise form of a claim made twice already: the deterministic solve of Chapter 1 and the probabilistic estimate of Chapter 3 are one computation. There is one set of rows, one Jacobian, one iteration. Adding a sensor adds a row; adding a prior adds a row; softening the physics changes a weight. The deterministic model is the corner of the probabilistic one in which the extra rows are absent, and the two can never disagree about the physics because they share it.

6.2.1 Gauss–Newton against Newton

The Gauss–Newton matrix $\mathbf{J}_g^\mathsf{T}\boldsymbol{\Omega}\mathbf{J}_g$ is not the Hessian of C. Differentiating equation (6.4) twice gives

$$\nabla^2 C \;=\; \mathbf{J}_g^\mathsf{T}\boldsymbol{\Omega}\mathbf{J}_g \;+\; \sum_k w_k\, g_k(\mathbf{z})\, \nabla^2 g_k(\mathbf{z}), \tag{6.7}$$

the Gauss–Newton term plus a sum, over every row, of that row's residual times its own curvature. The second term vanishes when the rows are linear, since then $\nabla^2 g_k = \mathbf{0}$, and it is small when the residuals at the solution are small, whatever the curvature, which is the case for a physical model whose physics rows are tight and whose sensors are fitted to their accuracies. Dropping it costs nothing in the linear case, keeps the matrix positive definite in every case, since a sum of weighted outer products cannot be indefinite while $\nabla^2 g_k$ can, and makes the iteration a sequence of least-squares solves that never needs a second derivative. What it costs is the rate: with the second term present Newton's method converges quadratically regardless of the residuals, and without it Gauss–Newton converges quadratically only when they vanish and linearly otherwise, at a rate set by the size of the dropped term relative to the kept one. On the models of this book the residuals at the mode are the physics widths and the sensor accuracies, both small, and the examples of §6.5 and §6.6 show the quadratic tail; a model with a badly fitted sensor, one whose residual at the mode is many accuracies, is the case where the dropped term matters, and it is also the case where Chapter 12 says something is wrong with the model.

A singular $\boldsymbol{\Lambda}$ is not a numerical accident but a statement: some direction in $\mathbf{z}$ is constrained by no physics row, no prior and no sensor, and the posterior is flat along it. The remedy is a prior on a variable in that direction, and the factorization that fails names the variable; §6.6 shows it on the water loop. Chapter 12 treats this as identifiability.

6.3 Elimination is Cholesky

Chapter 5 eliminated a variable by integrating it out and found that the neighbors fill in. For a Gaussian the integral is a formula. Partition $\mathbf{z}$ into a block $\mathbf{x}$ to eliminate and a block $\mathbf{u}$ to keep, and partition $\boldsymbol{\Lambda}$ and $\boldsymbol{\eta}$ accordingly. Then

$$p(\mathbf{u}) \;\propto\; \int p(\mathbf{x},\mathbf{u})\,\mathrm{d}\mathbf{x} \;=\; \mathcal{N}\big(\,\cdot\,;\; \boldsymbol{\Lambda}_{uu} - \boldsymbol{\Lambda}_{ux}\boldsymbol{\Lambda}_{xx}^{-1}\boldsymbol{\Lambda}_{xu},\; \boldsymbol{\eta}_u - \boldsymbol{\Lambda}_{ux}\boldsymbol{\Lambda}_{xx}^{-1}\boldsymbol{\eta}_x\big) \tag{6.8}$$

in information form. The matrix $\mathbf{\Lambda}_{uu} - \mathbf{\Lambda}_{ux}\mathbf{\Lambda}_{xx}^{-1}\mathbf{\Lambda}_{xu}$ is the *Schur complement* of $\mathbf{\Lambda}_{xx}$. Its sparsity is that of $\mathbf{\Lambda}_{uu}$ plus every pair of kept variables that are both adjacent to the eliminated block, which is the fill of §5.5 written as a matrix.

Proposition 6.3 (Marginalization is the Schur complement)**.** *Equation* (6.8) *holds: the marginal of a Gaussian in information form over a block of variables is Gaussian with the Schur complement as precision and* $\boldsymbol{\eta}_u - \mathbf{\Lambda}_{ux}\mathbf{\Lambda}_{xx}^{-1}\boldsymbol{\eta}_x$ *as information vector. The Schur complement is positive definite when* $\mathbf{\Lambda}$ *is.*

Proof. Write the exponent of the joint as $-\frac{1}{2}(\mathbf{x}^\mathsf{T}\mathbf{\Lambda}_{xx}\mathbf{x} + 2\mathbf{x}^\mathsf{T}\mathbf{\Lambda}_{xu}\mathbf{u} + \mathbf{u}^\mathsf{T}\mathbf{\Lambda}_{uu}\mathbf{u}) + \boldsymbol{\eta}_x^\mathsf{T}\mathbf{x} + \boldsymbol{\eta}_u^\mathsf{T}\mathbf{u}$. For fixed $\mathbf{u}$ it is a quadratic in $\mathbf{x}$; complete the square with $\mathbf{m} = \mathbf{\Lambda}_{xx}^{-1}(\boldsymbol{\eta}_x - \mathbf{\Lambda}_{xu}\mathbf{u})$:

$$-\tfrac{1}{2}(\mathbf{x}-\mathbf{m})^\mathsf{T}\mathbf{\Lambda}_{xx}(\mathbf{x}-\mathbf{m}) + \tfrac{1}{2}\mathbf{m}^\mathsf{T}\mathbf{\Lambda}_{xx}\mathbf{m} - \tfrac{1}{2}\mathbf{u}^\mathsf{T}\mathbf{\Lambda}_{uu}\mathbf{u} + \boldsymbol{\eta}_u^\mathsf{T}\mathbf{u}.$$

Integrating over $\mathbf{x}$ removes the first term and contributes a constant, $\sqrt{(2\pi)^{n_x}/\det\mathbf{\Lambda}_{xx}}$, that does not depend on $\mathbf{u}$. Expanding $\frac{1}{2}\mathbf{m}^\mathsf{T}\mathbf{\Lambda}_{xx}\mathbf{m} = \frac{1}{2}(\boldsymbol{\eta}_x - \mathbf{\Lambda}_{xu}\mathbf{u})^\mathsf{T}\mathbf{\Lambda}_{xx}^{-1}(\boldsymbol{\eta}_x - \mathbf{\Lambda}_{xu}\mathbf{u})$ and collecting the terms in $\mathbf{u}$ gives the quadratic $-\frac{1}{2}\mathbf{u}^\mathsf{T}(\mathbf{\Lambda}_{uu} - \mathbf{\Lambda}_{ux}\mathbf{\Lambda}_{xx}^{-1}\mathbf{\Lambda}_{xu})\mathbf{u} + (\boldsymbol{\eta}_u - \mathbf{\Lambda}_{ux}\mathbf{\Lambda}_{xx}^{-1}\boldsymbol{\eta}_x)^\mathsf{T}\mathbf{u}$, which is the claim. For positive definiteness, take any $\mathbf{u} \neq \mathbf{0}$ and set $\mathbf{x} = -\mathbf{\Lambda}_{xx}^{-1}\mathbf{\Lambda}_{xu}\mathbf{u}$; then $(\mathbf{x},\mathbf{u})^\mathsf{T}\mathbf{\Lambda}(\mathbf{x},\mathbf{u}) = \mathbf{u}^\mathsf{T}(\mathbf{\Lambda}_{uu} - \mathbf{\Lambda}_{ux}\mathbf{\Lambda}_{xx}^{-1}\mathbf{\Lambda}_{xu})\mathbf{u}$, and the left side is positive because $\mathbf{\Lambda}$ is. □

Eliminating one variable at a time, in some order, is exactly what the Cholesky factorization $\mathbf{\Lambda} = \mathbf{L}\mathbf{L}^\mathsf{T}$ does: each step pivots on one diagonal entry and updates the trailing submatrix by the rank-one Schur complement. The nonzeros of $\mathbf{L}$ that are not nonzeros of $\mathbf{\Lambda}$ are the fill edges, the order of elimination is the order of the pivots, and the treewidth of $\mathbf{\Lambda}$'s graph is what bounds the largest dense block that $\mathbf{L}$ contains. Chapter 5's statements about cost are statements about the size of $\mathbf{L}$.

6.3.1 What the factorization costs

A column-oriented Cholesky factorization that finds c_j nonzeros below the diagonal in column j of $\mathbf{L}$ spends about $c_j(c_j+3)/2$ multiply-adds on that column: one square root, c_j divisions, and a rank-one update of the $c_j \times c_j$ trailing block that the column touches. The total is a sum over columns of a quadratic in the column counts, and the column counts are the fronts of Chapter 5's elimination. So the cost is set by the order, through the fill it creates, and Table 6.1 counts it on the book's two small models and on square grids, the graph of a grid network, where the difference between orders is large enough to see.

On the 1 600-node grid the minimum-degree order factors in 2.9×10^5 multiply-adds, four and a half times fewer than the natural order and more than two thousand times fewer than a dense factorization, and its factor holds 1.7 percent of the entries a dense one would. The ratio to dense grows with the size, since the dense cost grows as n^3 and the sparse one, for a grid, as $n^{3/2}$. The physical networks of Chapter 18, with their smaller elimination widths, do better still. This is the arithmetic behind Principle 5.4.

With $\mathbf{L}$ in hand:

- the MAP is two triangular solves, $\mathbf{L}\mathbf{v} = \boldsymbol{\eta}$ then $\mathbf{L}^\mathsf{T}\mathbf{z}^\star = \mathbf{v}$;
- the marginal variance of z_k is $(\mathbf{\Lambda}^{-1})_{kk}$, and it is obtained by solving $\mathbf{\Lambda}\mathbf{x} = \mathbf{e}_k$ with the same two triangular solves and reading x_k, never by forming $\mathbf{\Lambda}^{-1}$, which is dense. This is *selected inversion*, and it costs one solve per variable asked about;

Table 6.1: Nonzeros of the Cholesky factor and multiply-adds of the factorization, for the chain and the water loop under the orders of this chapter and for square grids of n nodes under the natural (row by row) order and a minimum-degree order, against a dense factorization of the same size.

model	order	nonzeros of $\mathbf{L}$	multiply-adds
chain, 6 variables	inputs first	14	25
	flows first	20	50
water loop, 7 variables	minimum degree	20	42
	flows first	23	55
grid 10×10, $n = 100$	natural	1 009	5.9×10^3
	minimum degree	656	2.9×10^3
	dense	5 050	1.7×10^5
grid 40×40, $n = 1\,600$	natural	63 895	1.3×10^6
	minimum degree	21 508	2.9×10^5
	dense	1 280 800	6.8×10^8

- the marginal of any subset $\mathbf{u}$ is the Schur complement (6.8), and when $\mathbf{u}$ is the set of ports of a region, that Schur complement is the region summarized to its neighbors. Chapter 8 is about that.

Two of the three items deserve their own statements, because later chapters lean on them.

Proposition 6.4 (Selected inversion and sampling)**.** *Let $\mathbf{\Lambda} = \mathbf{L}\mathbf{L}^\mathsf{T}$. Then the marginal variance of z_k is $\|\mathbf{L}^{-1}\mathbf{e}_k\|^2$, obtained by one forward triangular solve; and if $\boldsymbol{\xi}$ is a vector of independent standard normals, then $\mathbf{z} = \boldsymbol{\mu} + \mathbf{L}^{-\mathsf{T}}\boldsymbol{\xi}$ is a sample from $\mathcal{N}(\boldsymbol{\mu}, \mathbf{\Lambda}^{-1})$, obtained by one backward triangular solve.*

Proof. $(\mathbf{\Lambda}^{-1})_{kk} = \mathbf{e}_k^\mathsf{T}(\mathbf{L}\mathbf{L}^\mathsf{T})^{-1}\mathbf{e}_k = (\mathbf{L}^{-1}\mathbf{e}_k)^\mathsf{T}(\mathbf{L}^{-1}\mathbf{e}_k)$, the squared norm of the solution of $\mathbf{L}\mathbf{v} = \mathbf{e}_k$. For the sample, $\mathbf{z} - \boldsymbol{\mu} = \mathbf{L}^{-\mathsf{T}}\boldsymbol{\xi}$ is Gaussian with mean zero and covariance $\mathbf{L}^{-\mathsf{T}}\,\mathbf{I}\,\mathbf{L}^{-1} = (\mathbf{L}\mathbf{L}^\mathsf{T})^{-1} = \mathbf{\Lambda}^{-1}$. □

Both cost a triangular solve, which for a sparse factor is linear in the factor's nonzeros. So every marginal variance of a model with a million variables costs a million sparse solves, which is affordable, and a posterior sample costs one, which is how Chapter 10 draws samples from a Gaussian region without ever forming a covariance.

Proposition 6.5 (Conditioning is the other Schur complement)**.** *In information form, conditioning on the block $\mathbf{x} = \mathbf{x}_0$ leaves $\mathbf{u}$ Gaussian with precision $\mathbf{\Lambda}_{uu}$, unchanged, and information vector $\boldsymbol{\eta}_u - \mathbf{\Lambda}_{ux}\mathbf{x}_0$.*

Proof. Fix $\mathbf{x} = \mathbf{x}_0$ in the exponent of equation (6.1) and keep the terms in $\mathbf{u}$: $-\frac{1}{2}\mathbf{u}^\mathsf{T}\mathbf{\Lambda}_{uu}\mathbf{u} + (\boldsymbol{\eta}_u - \mathbf{\Lambda}_{ux}\mathbf{x}_0)^\mathsf{T}\mathbf{u}$ plus a constant. □

Marginalizing and conditioning are thus the two halves of one block elimination. Marginalizing $\mathbf{x}$ changes the precision on $\mathbf{u}$ by the Schur complement and needs a solve; conditioning on $\mathbf{x}$ leaves the precision alone and needs only a product. The asymmetry is the information form's whole convenience: the questions that Part IV asks most, "given these readings" and "given this intervention", are conditionings, and they are cheap.

Principle 6.1. *For a linear-Gaussian model the whole of inference is one sparse factorization. The mode, every marginal variance, every conditional and every region summary are solves against it.*

6.4 The Laplace posterior

When some rows are nonlinear the posterior is not Gaussian, and its exact marginals are integrals with no closed form. But the Gauss–Newton iteration (6.6) still converges to the mode $\mathbf{z}^\star$, and the matrix assembled to take the last step, $\mathbf{\Lambda}(\mathbf{z}^\star) = \mathbf{J}_g(\mathbf{z}^\star)^\mathsf{T}\mathbf{\Omega}\mathbf{J}_g(\mathbf{z}^\star)$, is the curvature of C at the mode. The *Laplace approximation* replaces the posterior by the Gaussian with that mode and that curvature,

$$p(\mathbf{z} \mid \mathbf{y}) \;\approx\; \mathcal{N}\big(\mathbf{z}^\star,\; \mathbf{\Lambda}(\mathbf{z}^\star)^{-1}\big). \tag{6.9}$$

It is exact when every row is linear, since then $\mathbf{\Lambda}$ does not depend on $\mathbf{z}$ and the posterior is the Gaussian it describes. Otherwise it is the second-order expansion of $\log p$ about its maximum, and it is accurate when the posterior is concentrated enough that the rows are nearly linear across its width. A closure that curves by a small fraction of its slope over one posterior standard deviation is nearly linear in that sense; a closure that saturates or switches regime inside the posterior's width is not, and Chapter 7 is about those.

The Laplace posterior costs nothing beyond the MAP: the matrix is already assembled. What it does not provide is a guarantee, and Chapter 12 supplies tests that probe its validity from the model itself. The worked example below supplies the simplest test, comparison with exact quadrature, in the one case where quadrature is affordable.

6.5 Worked example: the chain in matrix form

Take the soft pumping chain of Chapter 4 with both sensors: six unknowns, four physics rows at width 0.01 kg/s, two priors, two sensors, eight rows of $\mathbf{g}$ in all. Every number here is from the script `ch06_gaussian_chain.py` in the book's `scripts` directory.

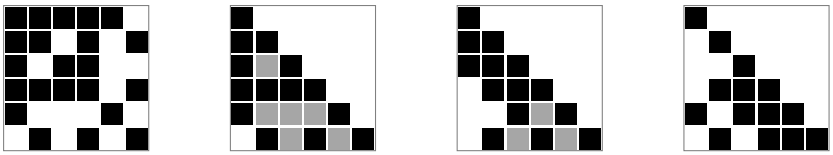
The Jacobian and the precision. Table 6.2 shows which variables each row of $\mathbf{g}$ reads. Form $\mathbf{\Lambda} = \mathbf{J}_g^\mathsf{T}\mathbf{\Omega}\mathbf{J}_g$ and mark its nonzeros:

$$\mathbf{\Lambda} \;\sim\; \begin{pmatrix} \bullet & \bullet & \bullet & \bullet & \bullet & \cdot \\ \bullet & \bullet & \cdot & \bullet & \cdot & \bullet \\ \bullet & \cdot & \bullet & \bullet & \cdot & \cdot \\ \bullet & \bullet & \bullet & \bullet & \cdot & \bullet \\ \bullet & \cdot & \cdot & \cdot & \bullet & \cdot \\ \cdot & \bullet & \cdot & \bullet & \cdot & \bullet \end{pmatrix} \qquad (m_{12},\, m_{2a},\, h_1,\, h_2,\, G,\, h_a). \tag{6.10}$$

The off-diagonal nonzeros are the eight edges T_1T_2, T_1m_{12}, T_2m_{12}, T_2m_{2a}, T_2T_a, $m_{12}m_{2a}$, $m_{12}G$, $m_{2a}h_a$, which is the Markov graph of Figure 3.1 read off Table 6.2: two variables are adjacent when some row has a mark in both columns. Proposition 6.1, by inspection.

Table 6.2: Rows of the stacked residual $\mathbf{g}$ for the chain with two sensors, and the variables each reads. The pattern of $\mathbf{J}_g$.

Row	Residual	m_{12}	m_{2a}	h_1	h_2	G	h_a
b_1	$m_{12} - G$	•				•	
b_2	$m_{2a} - m_{12}$	•	•				
c_{12}	$m_{12} - k(h_1 - h_2)$	•		•	•		
c_{2a}	$m_{2a} - k_2(h_2 - h_a)$		•		•		•
π_G	$G - 100$					•	
π_{h_a}	$h_a - 20$						•
ℓ_2	$h_2 - 72.0$				•		
ℓ_1	$h_1 - 93.0$			•			



Λ
14 in the lower loop

L, flows first
fill 6

L, potentials first
fill 3

L, inputs first
fill 0

Figure 6.2: The chain's precision $\mathbf{\Lambda}$ and its Cholesky factor $\mathbf{L}$ under three orderings, as nonzero patterns. Black entries of $\mathbf{L}$ are nonzeros of the lower triangle of the permuted $\mathbf{\Lambda}$; grey entries are fill. Inputs first has none; potentials first has three; flows first, the order in which the unknowns are listed, has six.

The MAP and the marginals. One solve of equation (6.5) gives the MAP: $m_{12} = m_{2a} = 104.63$, $h_1 = 92.97$, $h_2 = 72.04$, $G = 104.63$, $h_a = 19.73$. The two flows and the source agree to every digit because the physics rows are tight. The marginal standard deviations, by selected inversion against the Cholesky factor, are 2.86, 2.86, 0.47, 0.44, 2.86, 1.73 in the same order, and they agree with the diagonal of the dense inverse to four decimals, as they must. The Schur complement of $\mathbf{\Lambda}$ onto the two inputs (G, h_a) gives standard deviations 2.86 and 1.73 with correlation -0.979, which is the third column of Table 3.2. Chapter 3's hand calculation on the manifold and this chapter's sparse factorization of the soft model are, as Proposition 4.1 said, the same answer.

Fill under three orderings. The lower triangle of $\mathbf{\Lambda}$ has 14 nonzeros. Factor it in three orders and count the nonzeros of $\mathbf{L}$:

Elimination order	nonzeros of $\mathbf{L}$	fill
$m_{12}, m_{2a}, h_1, h_2, G, h_a$ (flows first, as listed)	20	6
$h_1, h_2, m_{12}, m_{2a}, G, h_a$ (potentials first)	17	3
$G, h_a, h_1, h_2, m_{12}, m_{2a}$ (inputs first)	14	0

Figure 6.2 draws the three factors. Inputs first creates no fill at all: G has one neighbor, h_a has two that are already adjacent, and so on down the list, each variable's neighbors already a clique when its turn comes. That is a perfect elimination order, and it exists because the Markov

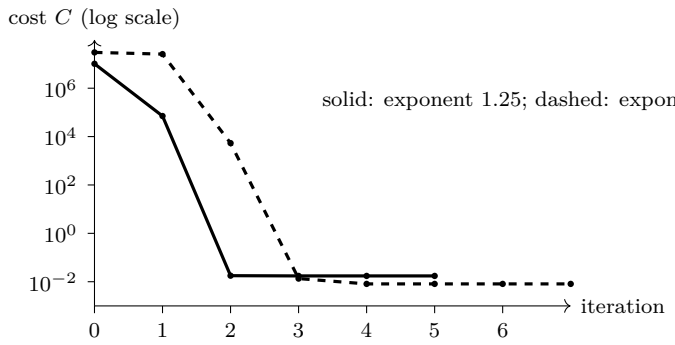


Figure 6.3: Gauss–Newton on the chain with the nonlinear the valve closure, from the cold start, one sensor: the cost at each iteration on a logarithmic scale, for the exponent 1.25 of the text (solid) and for exponent 4 (dashed). The first step removes seven orders of magnitude; the convergence is quadratic once the iterate is close.

graph of the chain is chordal. Flows first is the worst of the three: m_{12} has four neighbors, h_1, h_2, m_{2a}, G, of which only some pairs are adjacent, and eliminating it fills the rest. The fill is temporary in the sense of §5.5, since after every flow is gone the graph on the potentials is the network's, but $\mathbf{L}$ remembers it.

Flows first is also the order in which this book lists the unknowns, from Chapter 1 on, and there is no contradiction in that. The order in which variables are listed is for reading a model: the flows, then the potentials they connect, then the inputs. The order in which they are eliminated is for computing with it, and a sparse solver chooses that order separately, from the graph, before it does any arithmetic. On six variables none of this matters. On a million it decides whether the factorization fits in memory, and it is why a sparse solver spends effort choosing the order before it spends any on arithmetic.

A nonlinear closure. Replace the the valve closure by natural the valve, $m_{2a} = C\,(h_2 - h_a)^{1.25}$, with C chosen so that the two closures agree at the nominal operating point (104 kg/s at 52 metres). The row is now nonlinear in h_2 and h_a. Start Gauss–Newton from a deliberately poor guess, flows of 80, $h_1 = 60$, $h_2 = 50$, inputs at their prior means. With one sensor the cost falls from 1.0×10^7 to 7.1×10^4 to 0.018 in three iterations, with full steps throughout, and the step length is below 10^{-9} by the fifth. With two sensors it is the same story, one half-step near the end. Convergence is quadratic once the iterate is close, as the theory said. Figure 6.3 plots the cost per iteration for this closure and for a much harsher one, exponent 4, which takes seven iterations from the same start; the second script, `ch06_triangle_gaussian.py`, produces the figures of this section and the water loop that follows.

Table 6.3 compares the Laplace posterior at the mode with the exact posterior, computed by quadrature on an 801×801 grid over the two inputs in the input-space form of Chapter 4. With one sensor the Laplace mean of G is 0.12 kg/s below the exact mean, and the standard deviations agree to two decimals; the exact posterior is very slightly skewed, its mode sitting at the Laplace mean and its mean a little above. With two sensors the two agree to every digit shown. This is what a mildly nonlinear closure looks like: the row curves, but not enough across three kg/s of posterior width to matter.

Figure 6.4 draws the two densities, and adds the case of exponent 4: the Laplace spread is

Table 6.3: Natural-the valve closure: Laplace posterior at the Gauss–Newton mode against exact quadrature over (G, h_a).

Sensors		G [W]	h_a [m]
y_2 only	Laplace	103.50 ± 7.10	20.20 ± 2.81
	exact	103.62 ± 7.09	20.17 ± 2.81
y_2, y_1	Laplace	104.69 ± 3.02	19.76 ± 1.53
	exact	104.69 ± 3.02	19.76 ± 1.53

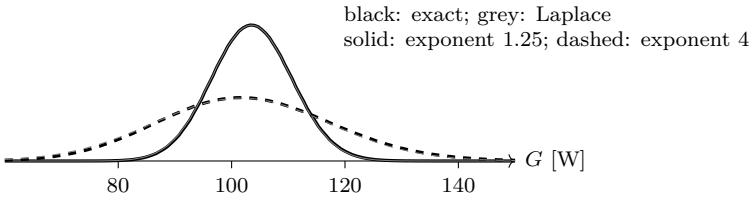


Figure 6.4: The marginal posterior density of the source with one sensor, exact by quadrature (black) and Laplace (grey), for the natural-the valve exponent 1.25 (solid) and for exponent 4 (dashed). At 1.25 the two are indistinguishable; at 4 the exact density is visibly skewed and its mean sits one kg/s above the Laplace mode, while the spreads still agree to one percent.

still within one percent of the exact one, 15.3 against 15.2 kg/s, but the exact density is skewed and its mean is a kg/s above the mode. The spread of a posterior is the first thing Laplace gets right and the skew the first thing it misses; Chapter 12 gives a probe that detects the skew from the model itself.

One number in the table deserves a physical reading. With one sensor the source's spread is 7.1 kg/s here against 5.8 in the linear model. The nonlinear closure has a smaller slope at the operating point: with $Q \propto \Delta T^{1.25}$, a kg/s of source moves h_2 by $\Delta T/(1.25\,G) \approx 0.4$ metres rather than 0.5. The sensor's lever arm on the source is shorter, so the same reading constrains it less. The Laplace posterior reports that correctly because $\mathbf{\Lambda}$ at the mode carries the local slope.

6.6 Second worked example: the water loop in matrix form

The water loop with uncertain demands and one flow meter, the model of §3.5, in the same matrix form: seven unknowns, the two heads, three flows and two demands; eight rows of $\mathbf{g}$, five physics rows at width 0.001, two demand priors and the meter. Table 6.4 gives the pattern of $\mathbf{J}_g$. The precision has eleven off-diagonal nonzeros, the eleven edges of the Markov graph: the seven of Figure 2.6 plus the four that the balances draw between the loads and the flows they balance. One solve gives the MAP, $q_2 = -1.421$, $q_3 = -0.460$, $m_{12} = 1.101$, $m_{13} = 0.780$, $m_{23} = -0.320$, which is the second column of Table 3.3, and selected inversion against the Cholesky factor gives the spreads 0.136, 0.269, 0.020, 0.135, 0.134, agreeing with the dense inverse to 10^{-12}. The condition number is 3.8×10^7, the value the sweep of Chapter 4 found at this width.

Fill. The lower triangle of $\mathbf{\Lambda}$ has 18 nonzeros. Angles first gives a factor with 22, four fill entries; flows first 23, five; the demands first, then the heads, then the flows, 20, two, which is what the minimum-degree heuristic also finds. The loads are the leaves of this graph, each read

Table 6.4: Rows of $\mathbf{g}$ for the water loop with uncertain demands and a meter on pipe 1–2, and the variables each reads. The pattern of $\mathbf{J}_g$.

Row	Residual	h_2	h_3	m_{12}	m_{13}	m_{23}	q_2	q_3
b_2	$m_{12} - m_{23} + q_2$			•		•	•	
b_3	$m_{13} + m_{23} + q_3$				•	•		•
c_{12}	$m_{12} + Bh_2$	•		•				
c_{13}	$m_{13} + Bh_3$		•		•			
c_{23}	$m_{23} - B(h_2 - h_3)$	•	•			•		
π_{q_2}	$q_2 + 1.5$						•	
π_{q_3}	$q_3 + 0.5$							•
ℓ_{12}	$m_{12} - 1.10$			•				

Table 6.5: Observability of the water loop's demands: the ratio of the smallest to the largest eigenvalue of the precision, and the posterior of the loads, with and without load priors, for one and two meters.

priors and meters	eigenvalue ratio	q_2	q_3
no priors; m_{12}	8.7×10^{-19}	unobservable	unobservable
no priors; m_{12}, m_{13}	6.9×10^{-7}	-1.260 ± 0.045	-0.780 ± 0.045
no priors; m_{12}, m_{23}	1.0×10^{-6}	-1.270 ± 0.028	-0.760 ± 0.045
load priors; m_{12}	2.6×10^{-8}	-1.421 ± 0.136	-0.460 ± 0.269

by one balance and its prior, and eliminating leaves first never fills; the heads then have two or three neighbors that are nearly cliques already. Flows first is again not the cheapest route, for the reason Chapter 5 counted: a flow's neighbors are the potentials at its ends and the other flows at both its junctions, and they are not adjacent to one another until the flow joins them.

Observability. Remove the load priors and keep the one meter. The precision is then singular: its smallest eigenvalue is 10^{-18} of its largest, and the eigenvector belonging to it is the direction $\Delta q_2 = -0.5$, $\Delta q_3 = 1$, with the flows on links 1–3 and 2–3 each moving by -0.5 and m_{12} not at all. That is the shift of demand from junction 2 to junction 3 that leaves the metered flow unchanged, since $m_{12} = -(2q_2 + q_3)/3$; the meter cannot see it, no prior pins it, and the posterior is flat along it. The factorization fails, and the vector that makes it fail names the unobservable combination. Table 6.5 lists the cures. A second meter on pipe 1–3 or on pipe 2–3 reads a different combination of the loads and restores a condition number near 10^6, with the demands known to 0.03 to 0.05 kg/s from the meters alone; the load priors restore it too, at the price of a condition number of 4×10^7 and of loads known only to 0.14 and 0.27, because a prior pins the unseen direction with its own width and no more. Observability in the power engineer's sense is the nonsingularity of this matrix with the priors removed.

Sampling from the factor. Proposition 6.4 turns the Cholesky factor into a sampler: 20 000 draws of $\boldsymbol{\mu} + \mathbf{L}^{-\mathsf{T}}\boldsymbol{\xi}$, one backward solve each, have sample spreads 0.137 and 0.270 for the loads against the exact 0.136 and 0.269, and a sample correlation of -0.976, the exact value. Figure 6.5 draws four hundred of them with the exact ellipses. The samples answer questions that the mean and covariance answer only with more algebra, and any question, including a nonlinear one: the probability that the boundary flow exceeds 2.1 kg/s is 0.0570 from the samples and

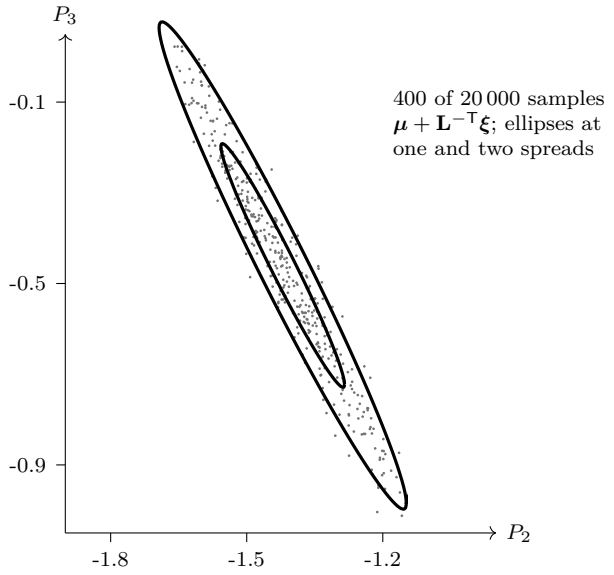


Figure 6.5: Four hundred posterior samples of the triangle's two loads, drawn from the Cholesky factor of the precision, with the exact one- and two-standard-deviation ellipses. The samples fill the ellipses in the proportions a Gaussian gives, because a triangular solve against the factor turns independent standard normals into draws with exactly the posterior covariance.

0.0574 exactly, and the probability that pipe 2–3 carries more than half a kilogram a second toward junction 2 is 0.089. Chapter 10 uses exactly this sampler inside a region.

The closure's true exponent. Everything above used the linearized pipe of Example 1.2. Put the real law back, $\Delta h = K\, m\, |m|^{0.852}$, and load the network heavily, with $k = 2.5$ so that the heads are large enough for the exponent to matter. Choose $K = 0.3508$ so that the nonlinear pipe loses exactly the same head as the linear one at the nominal flow of 1.167 kg/s, namely 0.4667 m: what is left between the two models is the curvature and nothing else.

Gauss–Newton from a flat start, heads zero and demands at their priors, reduces the cost from 3.7×10^5 to 4.7×10^3 to 6.2 to 0.019 in four iterations, full steps throughout, and converges by the seventh. The mode has $h_2 = -0.4182$ m where the linear pipe at the same flow would have lost 0.4399, a difference of 4.9 percent. The Laplace posterior of the demands is $q_2 = -1.549 \pm 0.167$ and $q_3 = -0.532 \pm 0.251$ kg/s; the exact posterior, by quadrature over the two demands with the hard Hazen–Williams physics solved by Newton's method at each grid point, is -1.543 ± 0.169 and -0.523 ± 0.252. The linearized model at the same conductance gives -1.421 ± 0.136 and -0.460 ± 0.269.

Two things follow, and the second is the one to carry forward. The Laplace posterior tracks the exact one to six thousandths of a kilogram a second in the mean and two thousandths in the spread: the approximation of this chapter survives the real exponent easily at this loading. But the *linearized* model does not. Five percent of curvature in the closure moved the demands' posterior by 0.13 and 0.07 kg/s, half a prior spread on the first, and narrowed one spread while widening the other. That is a much larger penalty than a mildly nonlinear domain would pay, and it is why Chapter 8 onward and all of Part VI carry the exponent rather than linearizing: in water the linearization is a device for making a textbook example checkable by hand, not a

modelling choice.

6.7 Filters, fields, and state estimators

Three things the reader may already know are this chapter under other names.

The *Kalman filter* is the case in which the variables are the state of a system at successive times, the physics rows are the transition from each time to the next, and the sensors read each time's state. The precision $\mathbf{\Lambda}$ is block-tridiagonal in time. Eliminating the blocks forward in time is the filter's prediction and update; eliminating backward is the smoother. Chapter 16 unrolls the graph and says so at length.

A *Gaussian Markov random field* is a Gaussian whose precision is sparse, with the sparsity read as a graph. That is equation (6.3) with Proposition 6.1. The literature on such fields is the literature on this chapter's objects.

Water-network state estimation is this chapter with the head-loss equations as the physics rows, the junction heads and pipe flows as the unknowns, meters and gauges as the sensors, and, in its classical form, no priors and hard physics. The weighted least-squares iteration it uses is equation (6.6), and its bad-data detection is the objective test that §4.5 previewed and Chapter 12 develops. Power-system state estimation is the same chapter again with the AC power flow equations in place of the head-loss ones. The framework of this book adds soft physics, priors on the demands — which in water is not a refinement but a necessity, since a distribution network's demands are almost never metered in real time — and the multi-domain graph; the linear algebra is the same, and Chapter 21 checks this construction against an independent weighted-least-squares optimizer on two hundred and eleven unknowns.

6.8 In practice

Assemble in information form and never invert. Build $\mathbf{\Lambda}$ by adding one rank-one term per factor, factor it once, and answer every question by a solve against the factor: the mode, the variance of any variable by selected inversion, the summary of any region by a Schur complement. A dense inverse of a sparse precision is dense, and forming it is the commonest way to turn a fast model into a slow one.

Let the solver choose the order. A fill-reducing ordering found by a minimum-degree or nested-dissection heuristic is nearly always better than any order a modeler writes by hand, and the chapter's counts show why: on the chain, inputs first has no fill and flows first has six; on the triangle, the heuristic finds the least. The exception is when the order is chosen for reuse, region interiors before ports, which Chapter 8 explains.

Scale the rows. A precision assembled from rows in mixed units, kg/s and metres and per-unit, has a condition number set by the ratio of the largest weight to the smallest, and the tight physics rows dominate. Equilibrate the rows as Chapter 4 described, and check the condition number after assembly; a value near 10^{10} means the physics widths are too tight for the arithmetic.

Start Gauss–Newton from the physics. A cold start converges in the examples of this chapter, but a start from the deterministic solve at the prior means costs one solve and removes the backtracking. Watch the step length: a run of half-steps means the linearization is poor

across the current iterate's neighborhood, and the Laplace posterior at the end deserves the checks of Chapter 12.

Report the Laplace posterior with its slope. The posterior spread of an input under a nonlinear closure is set by the closure's slope at the mode, as the natural-the valve example showed. When the spread differs from a linear model's, the difference should be explained by that slope, and if it cannot be, the mode or the linearization is suspect.

Test Laplace where you can. On any model with two or three uncertain inputs, quadrature over the inputs in input-space form is cheap and exact. Run it once on a representative case before trusting the Laplace posterior on the full model; the exponents 1.25 and 4 of Figure 6.4 show what a mild and a severe nonlinearity look like.

6.9 What is missing

The chapter assumed every variable continuous and every row at worst mildly nonlinear. A discrete availability, a closure that switches regime, a bounded parameter, or a strongly curved law breaks one of the two, and Chapter 7 says which of the machinery survives each. And it took the elimination order as given; organizing it by physical regions and ports, so that each region's Schur complement is computed once and reused, is Chapter 8.

Notes and further reading

A Gaussian with a sparse precision matrix whose pattern is read as a graph is a Gaussian Markov random field, and Rue and Held [80] develop everything in §6.1 and §6.3, including the information form, the Schur complement as marginalization, and the use of a sparse Cholesky factor for sampling and for marginal variances. Selected inversion, computing chosen entries of the inverse from the factor without forming the inverse, is Erisman and Tinney's [21], and it was invented for power networks. Sparse Cholesky, fill-reducing orderings and their graph theory are in Davis [16] and Duff, Erisman and Reid [19]; the first fill-reducing ordering for network equations is Tinney and Walker's [92].

Gauss–Newton with backtracking for nonlinear least squares, its descent property, and its quadratic convergence under small residuals are in Nocedal and Wright [65]. The Laplace approximation is classical; MacKay [56] and Bishop [9] present it in the form used here, as the Gaussian at the mode with the Hessian's curvature, and discuss when it is trusted. Proposition 6.2 is the book's statement of a fact that every implementation of weighted least squares over physics residuals relies on implicitly.

The three instances of §6.7 have their own literatures. The Kalman filter is Kalman's [38]; Särkkä and Svensson [82] give the modern treatment, including the view of filtering and smoothing as Gaussian inference on a chain, which is how Chapter 16 presents it. Power-system state estimation as weighted least squares over the power-flow equations is Schweppe's [83]; Monticelli [64] and Abur and Gómez Expósito [1] cover the Gauss–Newton iteration, sparse factorization, observability (the singular $\mathbf{\Lambda}$ of §6.2) and bad-data detection. Chavali and Nehorai [14] run the same estimation as message passing on a factor graph. Factor graphs for large sparse Gaussian and nearly-Gaussian estimation, with the same emphasis on the elimination order and the Cholesky factor, are the subject of Dellaert and Kaess [17].

Summary

- In information form, multiplying Gaussian factors is adding matrices. Every factor is a rank-one term on its scope; the posterior precision is $\mathbf{\Lambda} = \mathbf{J}_g^\top \mathbf{\Omega} \mathbf{J}_g$, and its sparsity is the Markov graph.
- The MAP solves $\mathbf{\Lambda}\mathbf{z}^\star = \boldsymbol{\eta}$; for nonlinear rows, Gauss–Newton with backtracking. With no priors and no sensors the step is Newton's step for the physics: determinism is a corner of inference.
- Elimination is Cholesky; fill is fill; the Schur complement is the marginal. Marginal variances come from solves against the factor, never from a dense inverse.
- The Laplace posterior is the Gaussian at the mode with the assembled curvature: exact for linear rows, accurate when rows are nearly linear across the posterior's width.
- On the chain: the precision pattern is the Markov graph by inspection; inputs-first elimination has zero fill, flows-first six; a natural-the valve closure converges in three Gauss–Newton steps from a cold start and its Laplace posterior matches quadrature to two decimals, and still to one percent in spread at exponent 4.
- On the triangle: eleven Markov edges, the MAP of Chapter 3 in one solve, selected inversion to 10^{-12}, leaves-first the cheapest order; a sine closure at heavy loading converges in five iterations and its Laplace posterior matches quadrature to a thousandth.

Exercises

Exercise 6.1. Show that the product of $\mathcal{N}(\boldsymbol{\mu}_1, \mathbf{\Sigma}_1)$ and $\mathcal{N}(\boldsymbol{\mu}_2, \mathbf{\Sigma}_2)$, as functions of $\mathbf{z}$, is proportional to a Gaussian with precision $\mathbf{\Sigma}_1^{-1} + \mathbf{\Sigma}_2^{-1}$ and information vector $\mathbf{\Sigma}_1^{-1}\boldsymbol{\mu}_1 + \mathbf{\Sigma}_2^{-1}\boldsymbol{\mu}_2$. Then derive the posterior of the worked example of §3.4 in one line from this rule.

Exercise 6.2. Proposition 6.3 gives the marginal in information form. Show that in covariance form the marginal of $\mathbf{u}$ is simply the block $\mathbf{\Sigma}_{uu}$ of the joint covariance, and hence that $(\mathbf{\Lambda}_{uu} - \mathbf{\Lambda}_{ux}\mathbf{\Lambda}_{xx}^{-1}\mathbf{\Lambda}_{xu})^{-1} = \mathbf{\Sigma}_{uu}$: the Schur complement of the precision is the inverse of a block of the covariance. Which form is cheaper to compute when $\mathbf{u}$ is small and $\mathbf{x}$ is large and sparse?

Exercise 6.3. For the chain's $\mathbf{\Lambda}$ in equation (6.10), carry out the inputs-first elimination symbolically, one variable at a time, and verify at each step that the eliminated variable's neighbors are already mutually adjacent. Then do the same for flows first and list the six fill entries.

Exercise 6.4. The water loop with uncertain demands (Figure 3.2, $a_{23} = 1$ fixed) and one flow meter is linear-Gaussian. Write its $\mathbf{g}$, its $\mathbf{J}_g$, and its $\mathbf{\Lambda}$, and find an elimination order with the least fill. Compare with the nodal form of Figure 2.5.

Exercise 6.5. Replace the natural-the valve exponent 1.25 by 2 and by 4 in the script and rerun the Laplace-versus-quadrature comparison with one sensor. The spreads agree to within one percent even at exponent 4; what does change, and why does a single coarse sensor keep Laplace honest where Chapter 12 will find a precise one does not? Relate the answer to the curvature of the closure across the posterior width.

Solutions to selected exercises

Exercise 6.2. In covariance form the joint is $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ and the marginal of a block is obtained by dropping the other coordinates: $\mathbf{u} \sim \mathcal{N}(\boldsymbol{\mu}_u, \boldsymbol{\Sigma}_{uu})$, because a marginal of a Gaussian is Gaussian with the corresponding sub-vector and sub-block. Proposition 6.3 gives the same marginal with precision $\boldsymbol{\Lambda}_{uu} - \boldsymbol{\Lambda}_{ux}\boldsymbol{\Lambda}_{xx}^{-1}\boldsymbol{\Lambda}_{xu}$, so that precision is $\boldsymbol{\Sigma}_{uu}^{-1}$: the Schur complement of the precision is the inverse of the covariance's block, which is the block-inverse identity read as a statement about marginals. When $\mathbf{u}$ is small and $\mathbf{x}$ is large and sparse, the Schur complement is the cheaper route: it needs one sparse factorization of $\boldsymbol{\Lambda}_{xx}$ and $|\mathbf{u}|$ solves against it, whereas $\boldsymbol{\Sigma}_{uu}$ by way of the full covariance needs the dense inverse of $\boldsymbol{\Lambda}$. For a region of a thousand interior variables and six ports that is six sparse solves against a dense million-entry inverse, and it is why Chapter 8 computes region messages as Schur complements.

Exercise 6.3. Inputs first: eliminate G, whose only neighbor is m_{12}, trivially a clique; then h_a, whose neighbors h_2 and m_{2a} are adjacent through c_{2a}; then h_1, neighbors h_2 and m_{12}, adjacent through c_{12}; then h_2, neighbors m_{12} and m_{2a}, adjacent through b_2; then m_{12}, neighbor m_{2a}; then m_{2a}. At every step the neighbors were already a clique, so no fill: a perfect elimination order, and the factor has the fourteen nonzeros of the lower triangle and no more. Flows first: eliminating m_{12}, with neighbors h_1, h_2, m_{2a}, G, fills $T_1 m_{2a}$, $T_1 G$, $T_2 G$ and $m_{2a} G$; eliminating m_{2a}, now with neighbors h_1, h_2, G, h_a, fills $T_1 T_a$ and $G T_a$; the remaining four are then a clique and nothing more fills. Six fill entries, the number the script counts, and every one of them joins a pair that the physics never related directly: a flow's neighbors on both sides of it, tied together only because the flow between them was removed.

Exercise 6.1. As a function of $\mathbf{z}$, the logarithm of $\mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ is $-\frac{1}{2}\mathbf{z}^\mathsf{T}\boldsymbol{\Sigma}_i^{-1}\mathbf{z} + \mathbf{z}^\mathsf{T}\boldsymbol{\Sigma}_i^{-1}\boldsymbol{\mu}_i$ plus a constant. The logarithm of the product is the sum, $-\frac{1}{2}\mathbf{z}^\mathsf{T}(\boldsymbol{\Sigma}_1^{-1} + \boldsymbol{\Sigma}_2^{-1})\mathbf{z} + \mathbf{z}^\mathsf{T}(\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_2^{-1}\boldsymbol{\mu}_2)$ plus a constant, which is the logarithm of a Gaussian with precision $\boldsymbol{\Sigma}_1^{-1} + \boldsymbol{\Sigma}_2^{-1}$ and information vector $\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_2^{-1}\boldsymbol{\mu}_2$: in information form, multiplying Gaussians adds their parameters. For the worked example of §3.4, work in the plane of the two inputs. The prior has precision $\mathrm{diag}(1/400,\ 1/9)$ and information $(100/400,\ 20/9)$. The reading $h_2 = 72$ with accuracy 0.5 reads $h_2 = h_a + G/2$, the linear function $\mathbf{h}^\mathsf{T}(G, h_a)$ with $\mathbf{h} = (\frac{1}{2}, 1)$, so it contributes precision $4\,\mathbf{h}\mathbf{h}^\mathsf{T}$ and information $4 \cdot 72\,\mathbf{h}$. In one line, the posterior has precision $\left(\begin{smallmatrix} 1.0025 & 2 \\ 2 & 4.1111 \end{smallmatrix}\right)$ and information $(144.25,\ 290.22)$, whose solution is $G = 103.7 \pm 5.82$ and $h_a = 20.16 \pm 2.87$, the second column of Table 3.2.

Exercise 6.4. In row form the variables are the two heads, the three flows and the two demands. The rows are the two balances, $m_{12} - m_{23} + q_2$ and $m_{13} + m_{23} + q_3$; the three closures, $m_{12} + Bh_2$, $m_{13} + Bh_3$ and $m_{23} - B(h_2 - h_3)$; the two demand priors; and the meter on m_{12}. Each row's variables form a clique of $\boldsymbol{\Lambda}$'s pattern: $\{m_{12}, m_{23}, q_2\}$, $\{m_{13}, m_{23}, q_3\}$, $\{m_{12}, h_2\}$, $\{m_{13}, h_3\}$ and $\{m_{23}, h_2, h_3\}$, the priors and the meter adding only diagonal entries. Eliminate the demands first: each has two neighbours already joined by its balance row. Then eliminate m_{12}, whose neighbours h_2 and m_{23} are joined by the closure of link 2–3, and then m_{13} likewise. What remains, h_2, h_3 and m_{23}, is one clique. No step creates a fill entry, so the order is perfect, and the largest clique has three variables. In the nodal form of Figure 2.5 the flows are gone. The variables are h_2, h_3, q_2 and q_3; the balances read $q_2 = B(2h_2 - h_3)$ and $q_3 = B(2h_3 - h_2)$, each touching its demand and both heads; and the meter reads $m_{12} = -Bh_2$, touching h_2 alone.

Eliminating the demands first is again perfect, and what remains is the pair of heads. The nodal form has four variables instead of seven and the same largest clique. The row form costs more variables for the same width, and buys a model in which every physical statement is a row that can be loosened or replaced on its own.

Chapter 7

Beyond Gaussian: hybrid states and nonlinearity

Chapter 6 assumed that every variable is continuous and every row is linear, or nearly so. Real systems break both assumptions, and they break them in a small number of ways: a component is in service or it is not; a device saturates or switches regime; a law curves strongly; a parameter has a bound. This chapter takes the four in turn, says what each does to the posterior, and says which parts of the machinery survive. The short answer is that the graph survives entirely, the Gaussian machinery survives conditionally on the discrete part, and what is lost is the single Gaussian: the posterior becomes a mixture, and the question of how many components it has is the question that Part III is organized around. The chapter's worked example is the smallest possible instance, a pipe that may be shut out of service, detected from one flow meter.

7.1 Four ways out of the Gaussian world

A discrete variable. The availability $a_{23} \in \{0, 1\}$ of Figure 3.2. Its prior is a probability mass, not a density; the closure that reads it is linear in the continuous variables for each fixed value of a_{23} and is not a single linear row across both values.

A piecewise closure. A valve that is proportional until it saturates, a limiter, a diode, a breaker: $y = f_b(x)$ with the regime b selected by the state itself. The closure is linear or smooth within each regime and has a kink at the boundary between regimes.

A strongly nonlinear closure. Radiation as the fourth power of head, AC power flow with its sines and cosines, a pump curve. The row is smooth but curves enough across the posterior's width that the Laplace posterior of §6.4 is in question.

A bounded parameter. A conductance or an efficiency that must be positive, an availability that must lie in $[0, 1]$, a demand that cannot be negative. A Gaussian prior puts mass where the parameter cannot be.

Nothing in this list changes the factor graph. A discrete variable is a circle; a piecewise closure is a square; a bound is a unary factor of a non-Gaussian shape. The scopes, the Markov graph, the separators and the treewidth are exactly as Chapters 2 and 5 described them. What changes is the arithmetic inside the squares, and therefore what the posterior looks like.

7.2 Hybrid states: the mixture over branches

Take a model with one discrete variable a and continuous variables $\mathbf{z}$. Write the joint as a sum over the values of a:

$$p(\mathbf{z}, a \mid \mathbf{y}) \;\propto\; \mathbb{P}(a)\; p(\mathbf{z} \mid a, \mathbf{y})\; Z_a, \qquad Z_a = \int p(\mathbf{y}, \mathbf{z} \mid a)\, \mathrm{d}\mathbf{z}. \tag{7.1}$$

For each fixed a the model is one of Chapter 6's: every row is linear in $\mathbf{z}$, so $p(\mathbf{z} \mid a, \mathbf{y})$ is Gaussian with a precision $\mathbf{\Lambda}_a$ and a mode $\mathbf{z}_a^\star$ obtained by one sparse solve. Call this the *branch* a. The number Z_a is the *evidence* of the branch: the probability of the readings under it, integrated over everything uncertain. For a linear-Gaussian branch it has a closed form,

$$\log Z_a \;=\; -\,C_a(\mathbf{z}_a^\star) \;-\; \tfrac{1}{2} \log\det \mathbf{\Lambda}_a \;+\; \text{const}, \tag{7.2}$$

where C_a is the branch's objective (6.4) and the constant collects the factors' normalizers, which are the same for every branch with the same rows and widths. The first term rewards a branch that fits the data and the priors closely; the second penalizes a branch whose posterior is sharply concentrated, since a sharp posterior means the branch predicted a narrow range of readings and was lucky to be right. Together they are the Gaussian form of the trade-off between fit and flexibility.

The log-determinant hides one more term, and it is the one that decides whether branches can be compared at all.

Proposition 7.1 (The evidence of a branch)**.** *Let a branch have inputs* $\mathbf{u}$ *with prior* $p(\mathbf{u})$*, solved variables* $\mathbf{x}$ *determined by* m *linear physics rows* $\mathbf{F}_a(\mathbf{x}, \mathbf{u})$ *with square Jacobian* $\mathbf{J}_{x,a}$ *with respect to* $\mathbf{x}$*, every physics width* σ*, and readings* $\mathbf{y}$ *with likelihood* $p(\mathbf{y} \mid \mathbf{x}, \mathbf{u})$*. As* $\sigma \to 0$ *the evidence* Z_a *of the branch's soft factors satisfies*

$$\frac{Z_a}{(\sqrt{2\pi}\,\sigma)^m} \;\longrightarrow\; \frac{1}{|\det \mathbf{J}_{x,a}|} \int p(\mathbf{u})\, p(\mathbf{y} \mid X_a(\mathbf{u}), \mathbf{u})\, \mathrm{d}\mathbf{u}. \tag{7.3}$$

The integral is the predictive density of the readings under the branch, and it does not depend on how the physics rows are written. Branches whose physics rows differ must therefore be weighted by $|\det \mathbf{J}_{x,a}|\, Z_a$*, or equivalently by the predictive density; weighting them by* Z_a *alone weights each by the reciprocal of its own physics determinant.*

Proof. Fix $\mathbf{u}$ and integrate the product of the physics factors and the likelihood over $\mathbf{x}$. Substitute $\mathbf{r} = \mathbf{F}_a(\mathbf{x}, \mathbf{u})$; for linear physics $\mathrm{d}\mathbf{x} = \mathrm{d}\mathbf{r}/|\det \mathbf{J}_{x,a}|$ with a constant Jacobian, and $\mathbf{x} = X_a(\mathbf{u}) + \mathbf{J}_{x,a}^{-1}\mathbf{r}$. The physics factors are $\prod_k \exp(-r_k^2/2\sigma^2)$, whose integral over $\mathbf{r}$ is $(\sqrt{2\pi}\sigma)^m$ and which, divided by that, tends to $\delta(\mathbf{r})$. So the inner integral tends to $(\sqrt{2\pi}\sigma)^m\, p(\mathbf{y} \mid X_a(\mathbf{u}), \mathbf{u})/|\det \mathbf{J}_{x,a}|$, and integrating over $\mathbf{u}$ against the prior gives the claim. For nonlinear physics $\mathbf{J}_{x,a}$ depends on $\mathbf{u}$ and stays inside the integral, which is Remark 4.1. □

The determinant is not a nuisance constant. Scaling one closure row by a factor, writing $m_{23}/B - (h_2 - h_3)$ instead of $m_{23} - B(h_2 - h_3)$, changes $|\det \mathbf{J}_x|$ and hence Z_a by that factor and changes nothing physical; the predictive density is unchanged. And a branch that changes a closure, as an outage does, changes the determinant with it. On the water loop the physics determinant with pipe 2–3 in service is three times its value with the pipe shut, because the nodal matrices are $B\left(\begin{smallmatrix} 2 & -1 \\ -1 & 2 \end{smallmatrix}\right)$ and $B\,\mathbf{I}$, with determinants $3B^2$ and B^2. The soft-row evidence

understates the in-service branch threefold, and the worked example below shows what that does to an alarm.

The posterior over the discrete variable follows by Bayes' rule,

$$\mathbb{P}(a \mid \mathbf{y}) \;\propto\; \mathbb{P}(a)\, Z_a, \tag{7.4}$$

and the posterior of any continuous variable is the *mixture*

$$p(z_j \mid \mathbf{y}) \;=\; \sum_a \mathbb{P}(a \mid \mathbf{y})\, \mathcal{N}\!\left(z_j;\; (\mathbf{z}_a^\star)_j,\; (\mathbf{\Lambda}_a^{-1})_{jj}\right), \tag{7.5}$$

one Gaussian per branch, weighted by the branch's posterior probability. Its mean and variance are those of a mixture: the mean is the weighted mean of the branch means, and the variance is the weighted variance within branches plus the variance of the branch means between them.

Principle 7.1. *A discrete variable splits the model into branches. Within a branch the model is linear-Gaussian and everything in Chapter 6 applies; across branches the posterior is a mixture weighted by prior times evidence. The evidence is one number per branch, and it comes from the same factorization that gives the branch's mode.*

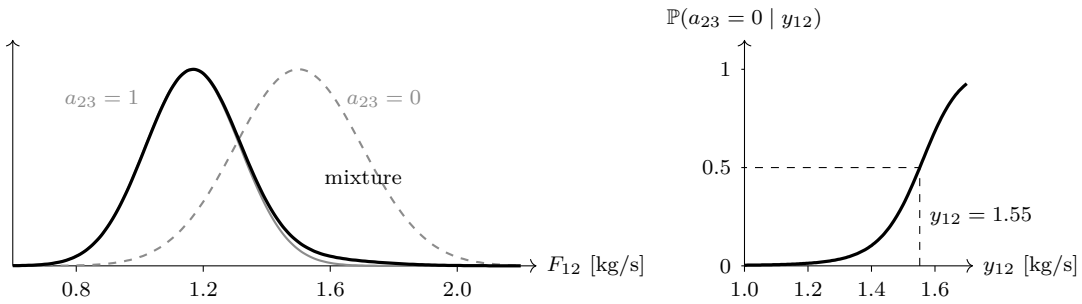
7.2.1 Worked example: a pipe that may be shut

Return to the water loop of Figure 3.2. The demands are uncertain, $q_2 \sim \mathcal{N}(-1.5, 0.2^2)$ and $q_3 \sim \mathcal{N}(-0.5, 0.2^2)$ kg/s, negative for a demand; the pipe conductances are $k = 10$; the physics rows have width 0.01; the pipe from junction 2 to junction 3 is shut with prior probability 0.05 — a valve closed for a repair and not recorded, which is the commonest model-form error a water utility carries; and one meter reads the flow on pipe 1–2 with accuracy 0.02. Every number is from the script `ch07_hybrid_triangle.py` in the book's `scripts` directory.

Before the reading. With the pipe open the metered flow is $m_{12} = -(2q_2 + q_3)/3$, which under the demand priors is 1.167 ± 0.149; the loop shares junction 2's demand with pipe 1–3 by way of pipe 2–3, which carries -0.333 ± 0.095. With the pipe shut, junction 2 is served by pipe 1–2 alone, $m_{12} = -q_2 = 1.500 \pm 0.200$, and pipe 2–3 carries nothing. Panel (a) of Figure 7.1 draws the prior predictive of the metered flow: the two branch Gaussians, weighted 0.95 and 0.05, and their sum. The outage branch is a small shoulder on the right of the main peak.

After the reading. Table 7.1 sweeps the reading from 1.10 to 1.60. Two things happen at once. Within each branch the posterior is Gaussian and the meter pins the flow to ± 0.02; the branch's estimate of q_2 is then whatever demand explains that flow under the branch's physics, -1.50 under the outage branch when the reading is 1.50, and -1.89 under the in-service branch, which needs a heavier demand at junction 2 to push that much flow down pipe 1–2 when part of it is being shared around the loop. Across branches the posterior probability of the outage rises from 0.006 at a reading of 1.10, where the reading sits on the in-service peak, to 0.31 at 1.50 and 0.68 at 1.60, where it sits on the outage branch's. Panel (b) draws the whole curve; it crosses one half at a reading of 1.55.

The weights come from Proposition 7.1, and the script checks them against the predictive density of the reading computed directly. Had the soft-row evidence been used without the



(a) prior predictive of the metered flow

(b) outage probability after the reading

Figure 7.1: Line-outage detection from one meter. (a) The prior predictive of the metered flow m_{12}: the in-service branch (solid grey) and the outage branch (dashed grey), each drawn at unit peak so that both are visible, and the weighted mixture (black), scaled to its own peak; in the mixture the outage branch is a shoulder of height about a twentieth of the main peak. (b) The posterior probability that the pipe is shut, as a function of the meter reading, with the branch evidence corrected by the physics determinant (Proposition 7.1); it crosses one half at $y_{12} = 1.55$.

Table 7.1: The water loop with an uncertain pipe and one meter: the outage posterior and the per-branch and mixture posteriors of the demand at junction 2 as the meter reading is swept.

y_{12}	$\mathbb{P}(a_{23} = 0 \mid y)$	$q_2 \mid a_{23} = 1$	$q_2 \mid a_{23} = 0$	$\mathbb{E}[q_2 \mid y]$	$\mathrm{sd}[q_2 \mid y]$
1.10	0.006	-1.42 ± 0.09	-1.11 ± 0.02	-1.42	0.10
1.17	0.010	-1.50 ± 0.09	-1.18 ± 0.02	-1.50	0.10
1.30	0.034	-1.66 ± 0.09	-1.30 ± 0.02	-1.64	0.11
1.40	0.104	-1.77 ± 0.09	-1.40 ± 0.02	-1.74	0.14
1.50	0.312	-1.89 ± 0.09	-1.50 ± 0.02	-1.77	0.20
1.60	0.685	-2.01 ± 0.09	-1.60 ± 0.02	-1.73	0.20

determinant, the in-service branch would have been understated threefold: the outage probability at 1.50 would have been reported as 0.58 instead of 0.31, at 1.60 as 0.87 instead of 0.68, and the crossover placed at 1.48 instead of 1.55. An alarm set at one half would have fired on readings between 1.48 and 1.55, which the correct posterior considers more likely normal than not.

What a single Gaussian would miss. Look at the last two columns at a reading of 1.50. The mixture's mean demand is -1.77 with a spread of 0.20. No branch believes that. The in-service branch says -1.89 ± 0.09 and the outage branch says -1.50 ± 0.02; the mixture mean lies between two modes, in a region of low posterior density, and its spread is mostly the distance between the modes, not uncertainty within either. An operator told "the demand is 1.67 give or take 0.2" would be told something that is not true under any hypothesis. The correct report is "either the pipe is shut and the demand is 1.50, or the pipe is open and the demand is 1.89, with odds of about 69 to 31 that it is in". This is what a Laplace posterior, which fits one Gaussian at one mode, cannot say, and it is the first of the ways it fails.

7.2.2 Many discrete variables

With k discrete variables there are 2^k branches, each a sparse Gaussian solve. For k up to ten or so that is affordable and exact. Beyond that, three things help, and Part III develops each: most branches have negligible posterior weight and can be pruned by their evidence; discrete variables that are separated in the graph can be summed out independently, which is the region argument of §5.7 with discrete separators; and the sum over branches can be sampled rather than enumerated. Chapter 15 applies all three to the question of which k pipes, out of hundreds, are most likely to be out together.

7.3 Piecewise closures and regimes

A piecewise closure is a discrete variable in disguise. Write the regime as b and the closure as $r_b(\mathbf{z}) = 0$; then the model is a mixture over b exactly as in §7.2, with one difference. The regime is not free: it is determined by the state. A valve is in its saturated regime when the demanded opening exceeds one, and in its proportional regime otherwise. So a branch b is *consistent* only if its own posterior mode puts the state in regime b, and inconsistent branches have no weight.

The deterministic version of this is the *active-set loop*: guess the regimes, solve, check each closure's regime against the solved state, switch the inconsistent ones, and repeat until no regime changes. It usually converges in a few passes, and when it does the solution is on a single branch. The probabilistic version does the same with posteriors: for each candidate assignment of regimes, solve the branch, check consistency at its mode, and keep the branches whose modes are consistent, weighted by evidence. Usually one survives. Two survive when the posterior straddles a regime boundary, and the posterior is then a mixture whose components sit on either side of a kink.

The kink matters for one thing more than any other. A gradient computed across it is wrong: the sensitivity of the state to an input jumps at the boundary, and a finite difference that spans the boundary averages two slopes that belong to different physics. Chapter 14 returns to this when it computes sensitivities, and it is the reason that chapter insists on differentiating within the active regime.

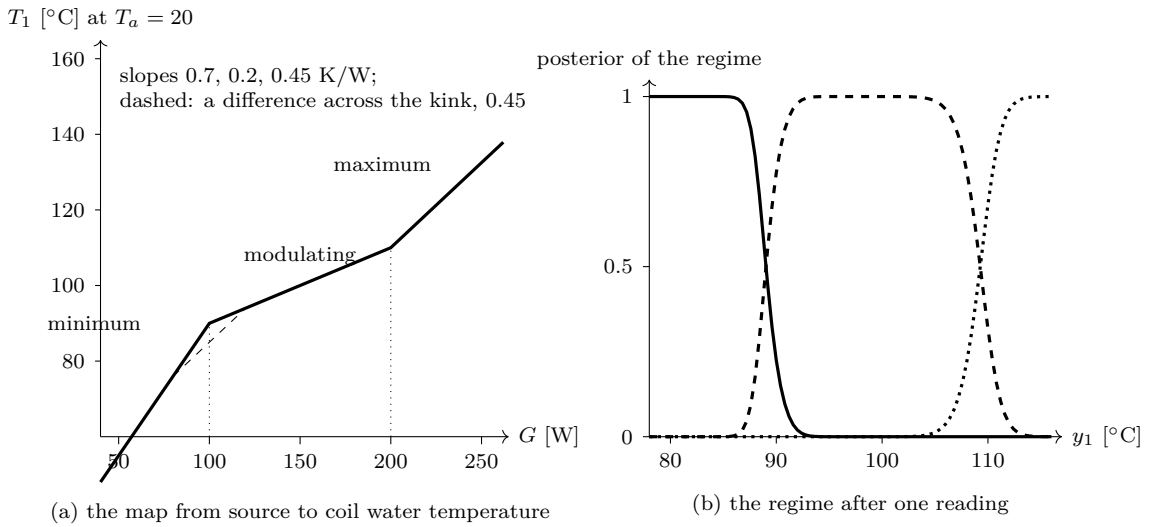
When the regimes join continuously, as a valve's do at saturation, the weight of a regime has an exact form that the active-set picture only approximates.

Proposition 7.2 (The weight of a regime)**.** *Let the physics be piecewise linear in the inputs* $\mathbf{u}$*: on region* $\mathcal{R}_b$ *of input space the solved variables are* $X_b(\mathbf{u})$*, the map of the linear branch* b*. Then the posterior probability of regime* b *is*

$$\mathbb{P}(b \mid \mathbf{y}) \;\propto\; p_b(\mathbf{y})\, \mathbb{P}_b(\mathbf{u} \in \mathcal{R}_b \mid \mathbf{y}), \tag{7.6}$$

where $p_b(\mathbf{y})$ *is branch* b*'s predictive density of the readings and* $\mathbb{P}_b(\,\cdot \mid \mathbf{y})$ *is branch* b*'s Gaussian posterior over the inputs. The posterior of any quantity within regime* b *is branch* b*'s posterior restricted to* $\mathcal{R}_b$*.*

Proof. The posterior over the inputs is $p(\mathbf{u})\, p(\mathbf{y} \mid X(\mathbf{u}))/p(\mathbf{y})$, and on $\mathcal{R}_b$ the true map X equals X_b. So the mass of regime b is $\int_{\mathcal{R}_b} p(\mathbf{u})\, p(\mathbf{y} \mid X_b(\mathbf{u}))\, \mathrm{d}\mathbf{u}$ divided by $p(\mathbf{y})$. Branch b's posterior is $p(\mathbf{u})\, p(\mathbf{y} \mid X_b(\mathbf{u}))/p_b(\mathbf{y})$ over all of input space, so the integral is $p_b(\mathbf{y})$ times that posterior's mass on $\mathcal{R}_b$. □



(a) the map from source to coil water temperature

(b) the regime after one reading

Figure 7.2: The pressure-reducing valve. (a) The discharge head as a function of the pumped flow at a trunk main of 20 m: continuous, with kinks where the valve reaches each end of its travel; the dashed chord is a central difference across the first kink. (b) The exact posterior probability of each regime as a function of the discharge-head reading: valve part shut (solid), modulating (dashed), wide open (dotted).

The proposition replaces the active-set test "is the branch's mode in its region" by a mass, "how much of the branch's posterior is in its region". The two agree when the posterior is far from every boundary and disagree near one, and the worked example below shows by how much.

7.3.1 Worked example: a pressure-reducing valve

Make the valve on the pumping chain's second section a *pressure-reducing valve*, the commonest control element in a water network. Its controller sets its opening to hold the junction at 70 m, and its conductance can range from 2 to 4 kg/s per metre — part shut at one end of its travel, wide open at the other. The closure then has three regimes, and every pressure-reducing valve in service is in one of them at any moment. With the valve at its minimum opening, $m_{2a} = 2(h_2 - h_a)$ and the junction sits below the setpoint: the valve cannot shut far enough to hold it up. With the valve modulating, $h_2 = 70$ and the conductance is whatever the controller needs, between the limits. With the valve wide open, $m_{2a} = 4(h_2 - h_a)$ and the junction sits above the setpoint: the valve is doing all it can and is no longer controlling. The first boundary is where the part-shut valve just holds the junction at 70, $G = 2(70 - h_a)$; the second is where the wide-open valve does, $G = 4(70 - h_a)$. The priors of Chapter 3, $G \sim \mathcal{N}(100, 20^2)$ kg/s and $h_a \sim \mathcal{N}(20, 3^2)$ m, put the first boundary through the prior mean: the prior gives 0.501 to the valve part shut, 0.499 to modulating, and nothing measurable to wide open, which needs 200 kg/s. A gauge on the discharge reads h_1 with accuracy 0.5 m. Every number is from `ch07_regimes.py`.

Knowing which regime a valve is in is not a modelling nicety. A pressure-reducing valve that has run out of travel has stopped controlling, and the network downstream of it is no longer at the head the operator believes it is at; the whole of this section is about reading that off a single gauge.

Table 7.2: The pressure-reducing valve after one reading of the discharge head: regime probabilities and the posterior of the source, exact by quadrature and by Proposition 7.2, and the modulating probability under the active-set shortcut.

	exact				proposition		shortcut
y_1	shut	mod	open	G [kg/s]	mod	G [kg/s]	mod
85	1.000	0.000	0.000	93.2 ± 4.2	0.000	93.2 ± 4.2	0.000
88	0.824	0.176	0.000	96.7 ± 4.3	0.179	96.7 ± 4.3	0.000
90	0.224	0.776	0.000	101.2 ± 2.7	0.780	101.2 ± 2.7	0.000
92	0.020	0.980	0.000	110.0 ± 2.4	0.981	110.0 ± 2.4	1.000
100	0.000	1.000	0.000	149.2 ± 2.5	1.000	149.2 ± 2.5	1.000
105	0.000	0.987	0.013	173.8 ± 2.5	0.987	173.8 ± 2.5	1.000
110	0.000	0.317	0.683	191.8 ± 6.5	0.319	191.8 ± 6.5	0.442
114	0.000	0.002	0.998	197.8 ± 6.4	0.002	197.8 ± 6.4	0.000

The map and its kinks. In the three regimes the discharge head is $h_a + 0.7G$, $70 + 0.2G$ and $h_a + 0.45G$. Panel (a) of Figure 7.2 draws the map at $h_a = 20$: continuous, with slopes 0.7, 0.2 and 0.45 metres per kg/s and kinks at 100 and 200 kg/s. A central difference across the first kink gives 0.45 metres per kg/s for steps of 20, 5 and 0.5 kg/s. It converges as the step shrinks, but to the average of the two slopes, not to either, and nothing in the number says that it belongs to no physics.

The weights. Table 7.2 sweeps the reading and computes the regime probabilities three ways: exactly, by quadrature over the two inputs with the piecewise physics; by Proposition 7.2, with each branch's in-region mass estimated from 400 000 samples of its Gaussian posterior; and by the active-set shortcut, which keeps a branch only if its mode lies in its region. The proposition matches quadrature to 0.004 in every regime probability and to a tenth of a kilogram a second in the source's mean and spread. The shortcut is right away from the boundaries and wrong near them. At a reading of 90, where the exact posterior gives 0.22 to the valve part shut and 0.78 to modulating, the shortcut reports it part shut with certainty: the modulating branch's mode sits on the edge of its region and is discarded, though most of that branch's posterior lies inside. At 110 it reports 0.44 modulating where the exact value is 0.32.

What the regime does to the sensor. In the modulating regime the discharge head is $70 + 0.2G$ whatever the trunk main does: the controller has absorbed the trunk, and the reading says nothing about it. At readings of 95 and 105 the posterior of the trunk head is its prior, 20.0 ± 3.0 m. The source is known to 2.5 kg/s there, which is the gauge's accuracy divided by the slope 0.2, while with the valve part shut the trunk's spread adds to the reading's and the source is known only to 4.2. What a sensor tells is a property of the regime, and the regime is a property of the state.

7.4 Strong nonlinearity: when the Laplace posterior holds and when it does not

Chapter 6 showed that a closure with exponent 1.25 left the Laplace posterior accurate to two decimals. Raise the exponent. Table 7.3 repeats the one-sensor comparison of Table 6.3 with the

Table 7.3: The chain with $m_{2a} = C\,(h_2 - h_a)^p$ and one sensor $y_2 = 72.0$: Laplace posterior against exact quadrature as the exponent grows. Skewness of the exact posterior of G in the last column.

p	Laplace G	exact G	Laplace h_a	exact h_a	skew
1.25	103.5 ± 7.1	103.6 ± 7.1	20.2 ± 2.8	20.2 ± 2.8	+0.04
2	102.9 ± 10.3	103.5 ± 10.3	20.3 ± 2.6	20.2 ± 2.6	+0.11
4	101.6 ± 15.3	102.7 ± 15.2	20.3 ± 2.0	20.3 ± 2.0	+0.14

the valve closure $m_{2a} = C\,(h_2 - h_a)^p$ at $p = 1.25$, 2 and 4, the constant C rescaled each time so that the closure passes through the nominal operating point.

The result is not what a reader expecting nonlinearity to break the approximation would predict. At $p = 4$, a fourth-power law, the Laplace standard deviation of the source is within one percent of the exact one and its mean is one kg/s low on a spread of fifteen. The reason is physical. A stiffer closure moves h_2 less per kg/s of source: the slope $\partial h_2/\partial G = \Delta T/(pG)$ falls from 0.4 metres per kg/s at $p = 1.25$ to 0.125 at $p = 4$. The sensor's lever arm on the source shortens, the posterior on G widens from ± 7 toward the prior's ± 20, and a posterior that is mostly prior is mostly Gaussian. The nonlinearity that might have distorted the posterior has instead removed the information that would have made the distortion visible. Curvature matters in proportion to how far the posterior extends across it, and here the extension and the curvature moved in opposite directions.

So strong nonlinearity by itself is a weaker enemy of the Laplace posterior than folklore suggests, at least for closures of this kind. The approximation fails in three recognizable situations, and a modeler should look for these rather than for large exponents.

Multiple modes. The outage example of §7.2.1: two branches, two modes, and a single Gaussian at either one misreports both the mean and the spread. Discrete structure is the usual cause; a closure that is symmetric in a variable, so that a sensor cannot distinguish a value from its negative, is another. Chapter 15's failure sets and the phase-head ambiguities of AC power flow are the cases that matter in practice.

A mode at a bound. When the posterior mode sits on a constraint, §7.5 below, the curvature at the mode describes a Gaussian half of which lies where the variable cannot go.

Curvature that survives concentration. When the data are informative enough to concentrate the posterior and the closure still curves within that concentrated width. Periodic closures with wide priors on heads are the standard example; so is a posterior that straddles a regime boundary. Here the mode is right and the curvature is not, and the marginal computed from $\mathbf{\Lambda}^{-1}$ can be wrong by a factor.

The tests that detect these from the model itself, without exact quadrature, are in Chapter 12: a comparison of the Laplace posterior's predicted spread of each residual with the actual residuals, and a probe that evaluates the objective at a few points along each posterior axis and checks that it is quadratic. The remedies, in increasing cost, are to reparameterize so that the closure is nearer to linear in the new variable, to integrate exactly in the one or two dimensions that are nonlinear and use Laplace for the rest, and to sample. Chapter 9 takes them up.

7.5 Bounded parameters

A Gaussian prior on a conductance $k \sim \mathcal{N}(5, 1^2)$ puts a probability of about 3×10^{-7} on $k < 0$, which is harmless. A prior $\mathcal{N}(1, 1^2)$ on an efficiency puts 0.16 on $\eta < 0$ and 0.5 on $\eta > 1$, which is not. Two remedies are standard.

Truncation. Multiply the Gaussian prior by the indicator of the admissible interval. The factor graph gains nothing; the prior's unary factor changes shape. The posterior mode is found by the same Gauss–Newton iteration with the step clipped at the bound, and if the unconstrained mode lies inside the interval nothing else changes. If the mode lies on the bound, the Laplace posterior is a Gaussian centered on the bound, half of it inadmissible; the honest report is the truncated Gaussian, whose mean is inside the interval and whose spread is smaller than $\mathbf{\Lambda}^{-1}$ says.

Reparameterization. Write the positive conductance as $k = e^{\kappa}$ and put the prior on κ; write the efficiency as $\eta = 1/(1 + e^{-\lambda})$ and put the prior on λ. The closure that read k now reads e^{κ}, which is one more nonlinearity in a row that is already treated by Gauss–Newton, and the bound has disappeared. This is usually the better choice: it keeps the machinery unchanged, and a Gaussian on κ is often a more honest prior for a quantity known to within a factor than a Gaussian on k. Its cost is that κ's posterior, mapped back to k, is skewed, and the Laplace posterior in κ must be transformed rather than read off.

The two remedies differ where the data are thin, and the chain shows where. Give the conductance a Gaussian prior $\mathcal{N}(3, 2^2)$, a contact conductance known only to lie somewhere between one and five, or a Gaussian prior on $\ln k$ with the same median and a spread of 0.6. With the two gauges of Chapter 3 the posterior of k is 4.92 ± 0.33 under the first and 4.92 ± 0.34 under the second: the data decide, and the bound never bites. Before the data the two priors differ sharply. Draw the prior predictive of the body head, $h_1 = h_2 + G/k$, by sampling the three inputs. Under the Gaussian prior 6.6 percent of the draws have $k \le 0$, and in every one of them flows uphill from the junction into the discharge; the fifth percentile of h_1 is -7 metres, and 3.4 percent of the draws put the discharge above 300 metres because k is near zero. Under the log-normal prior no draw is unphysical, and the fifth to ninety-fifth percentiles are 71 to 169 metres. Figure 7.3 draws both. A prior predictive is exactly what a Monte Carlo study of uncertain inputs computes, and a Gaussian prior on a positive parameter hands that study samples that are not physics.

7.6 What survives

It is worth listing what the four departures leave standing, because it is most of the book.

The factor graph, its Markov graph, its separators and its treewidth are untouched. Proposition 5.1 holds for any factors, and Proposition 5.3 is about the physics rows' scopes, which the departures do not change. Elimination and fill are the same graph operations; what changes is that eliminating a discrete variable is a sum and eliminating a continuous one is an integral, and a factor that has both kinds of variable in its scope must be represented as a table of Gaussians, one per discrete configuration. Such *conditional Gaussian* factors have their own algebra, and junction-tree inference with them is exact when the discrete variables are eliminated after the continuous ones in each clique; the rule and its limits are in the notes.

The Gaussian machinery survives per branch. Every branch is a sparse solve, every branch's evidence is its objective plus a log-determinant from the same factorization, and every branch's

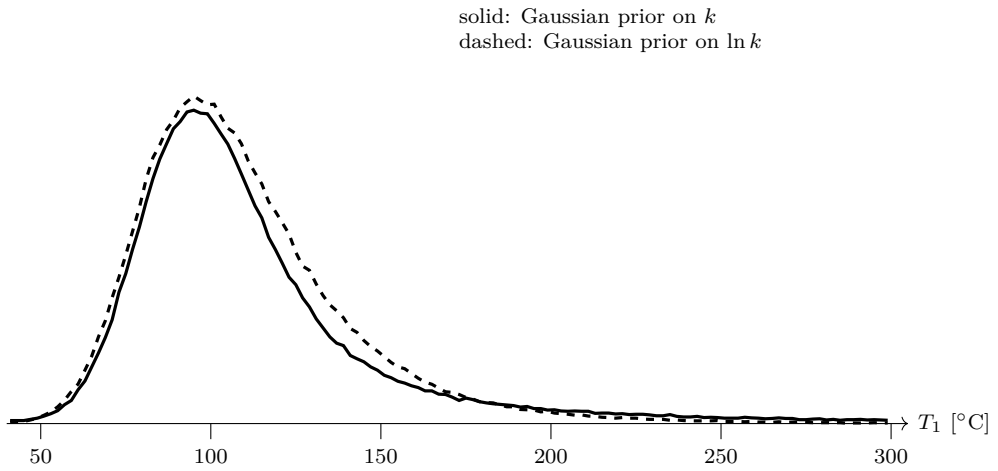


Figure 7.3: The prior predictive density of the discharge head under a Gaussian prior on the conductance (solid) and a Gaussian prior on its logarithm (dashed), both with median 3 kg/s per metre. The solid curve's heavy right tail comes from conductances near zero; its 6.6 percent of draws with negative conductance, all with the discharge colder than the junction, lie off the left of the figure.

marginals are selected inversions. The Laplace posterior survives within a branch under the conditions of §7.4.

What does not survive is the single Gaussian. The posterior is a mixture over branches and over regimes, and the honest summary of a mixture is its components and their weights, not its mean and variance. Where the number of branches is small the mixture is enumerated exactly; where it is large, Part III is about how to avoid enumerating it.

7.7 In practice

Compare branches by their predictive densities. When the branches differ in their physics, as an outage or a regime does, weight each by the predictive density of the readings, or by its soft-row evidence times its physics determinant. Never compare soft-row evidences alone; Proposition 7.1 says what that does, and the outage example shows it raising an alarm on a normal reading.

Report the components. A mixture's mean and variance are a summary that no branch believes when the weights are comparable. Report each branch's estimate and its weight, and collapse to one Gaussian only when one weight is near one.

Weight regimes by mass, not by mode. For a piecewise closure, Proposition 7.2 weights a regime by the part of its branch's posterior that lies in the regime's region. Keeping a branch because its mode is in its region is exact far from the boundaries and wrong near them, which is where the operator's questions usually are.

Differentiate within the regime. A finite difference across a kink converges to the average of two slopes. Take derivatives of the active regime's closure at the operating point, and if the

operating point is near a boundary, report both slopes.

Parameterize positive quantities by their logarithms. A Gaussian prior on a conductance, an efficiency or a demand is harmless once the data are in and wrong before they are, and the prior predictive is what a Monte Carlo study and a planning study compute. Put the prior on the logarithm, or the logit for a quantity in an interval.

Test Laplace where it can fail. Multiple modes, a mode on a bound, and curvature that survives concentration are the three failures; a large exponent is not one of them. Look for the three, and use the probes of Chapter 12 to check.

7.8 What is missing

Part II is complete. The model is a factor graph whose squares are soft physics, priors, sensors and failure factors; its posterior is a mixture of sparse Gaussians; its structure decides what is independent of what and what elimination costs. What has not been said is how to organize the computation when the graph is large: how to eliminate by regions and reuse the result (Chapter 8), what to do when exact elimination is too expensive (Chapter 9), how the factorized computation compares with sampling when both are costed in physics solves (Chapter 10), and how to fold in evidence as it arrives (Chapter 11). That is Part III.

Notes and further reading

Graphical models with both discrete and continuous variables, in which the continuous ones are Gaussian conditional on the discrete ones, are the conditional Gaussian models of Lauritzen and Wermuth [51]; exact propagation of probabilities, means and variances on a junction tree for such models, with the constraint on the elimination order mentioned in §7.6, is Lauritzen's [48]. The evidence of a branch as the integrated likelihood, its closed form for Gaussian models, and its use to compare hypotheses are treated by Kass and Raftery [41] under the name of Bayes factors, and by MacKay [56], whose account of the trade-off between fit and flexibility in equation (7.2) is the one followed here. That the soft-row evidences of branches with different physics must be corrected by the physics determinant, Proposition 7.1, is the coarea factor of Remark 4.1 appearing in model comparison; it is stated here because the soft formulation makes the uncorrected comparison easy to compute and wrong.

The accuracy of the Laplace approximation to posterior moments, with the error terms that explain why a nearly-Gaussian posterior is approximated well, is Tierney and Kadane's [91]; Bishop [9] gives the textbook account. The observation of §7.4, that a stiffer physical closure can make the posterior more Gaussian rather than less by weakening the sensor's lever arm, is the book's, and the exponent sweep is its evidence.

Line-outage detection from measurements is a standard problem in power-system operation, usually posed as a hypothesis test over a list of candidate outages; the treatment in §7.2.1 as a posterior over a discrete variable in the same graph as the state is the framework's, and Chapter 15 scales it up. The active-set loop of §7.3 is the standard treatment of piecewise constitutive laws in circuit and network simulators. Reparameterization of positive and bounded parameters by logarithms and logits is the common practice of Bayesian modeling [56].

Summary

- Four departures from the Gaussian world: a discrete variable, a piecewise closure, a strongly nonlinear closure, a bounded parameter. None changes the factor graph.
- A discrete variable splits the model into linear-Gaussian branches. Each has a mode, a precision and an evidence from one sparse factorization; the discrete posterior is prior times evidence; the continuous posterior is a mixture.
- Branches whose physics differ are compared by their predictive densities, which is the soft-row evidence times the physics determinant. On the triangle the uncorrected evidence understates the in-service branch threefold and would raise an outage alarm too early.
- One flow meter on the water loop detects a shut pipe: the outage probability rises from 0.006 to 0.68 across a reading sweep, and at the crossover the demand posterior is bimodal, which no single Gaussian can report.
- A piecewise closure is a discrete variable determined by the state. A regime's weight is its branch's predictive density times the branch posterior's mass in the regime's region, which matches quadrature on the valve to 0.004; keeping branches whose modes are in their regions fails near the boundaries. A finite difference across a kink converges to the average of two slopes.
- In the valve's modulating regime the sensor says nothing about the trunk main: what a reading tells depends on the regime.
- Raising a closure's exponent to four did not break the Laplace posterior, because the stiffer law shortened the sensor's lever arm and returned the posterior toward its Gaussian prior. Laplace fails at multiple modes, at a mode on a bound, and when curvature survives concentration.
- Bounded parameters: truncate, or reparameterize by a logarithm or a logit. Reparameterization is usually better: with data both give the same posterior, but a Gaussian prior on a positive conductance sends 6.6 percent of prior-predictive draws uphill.

Exercises

Exercise 7.1. Derive equation (7.2) by completing the square in the branch's objective and integrating the Gaussian, and identify the constant. Then show that for two branches with the same rows and widths but different row coefficients the constant cancels and the log-determinant does not: it contains $\log|\det \mathbf{J}_{x,a}|$, which Proposition 7.1 says must be added back.

Exercise 7.2. In the outage example, move the meter to pipe 2–3 instead of pipe 1–2. Compute the prior predictive of its reading under both branches and explain why this meter detects the outage from a single reading with near certainty, whatever the demands. Then move it to pipe 1–3 and explain why it is exactly as informative as the meter on pipe 1–2.

Exercise 7.3. Add a second uncertain pipe to the loop, pipe 1–3, so that there are four branches. Compute the prior probability of each, the predictive density of each at a reading $y_{12} = 1.50$, and the posterior. Which two branches can this meter not tell apart, and what does their posterior ratio equal?

Exercise 7.4. A valve closure is $Q = c\,u\,\Delta p$ for opening $u \in [0, 1]$ and $Q = c\,\Delta p$ for $u \geq 1$. Write the two regimes as rows, state the consistency condition for each, and construct a prior on

u and a sensor on Q for which both regimes are consistent at their modes. Draw the resulting posterior of u and mark the kink.

Exercise 7.5. Repeat the exponent sweep of Table 7.3 with the sensor accuracy improved from 0.5 to 0.05 metres, so that the posterior on G is concentrated. Find the exponent at which the Laplace standard deviation first errs by more than ten percent, and relate the answer to the third failure mode of §7.4.

Solutions to selected exercises

Exercise 7.1. For a fixed branch every factor is Gaussian in $\mathbf{z}$ with weight w_k, so the integrand of Z_a is $\prod_k (w_k/2\pi)^{1/2} \exp(-C_a(\mathbf{z}))$, with C_a the branch's quadratic objective. A quadratic equals its value at its minimum plus half its Hessian form about the minimum: $C_a(\mathbf{z}) = C_a(\mathbf{z}_a^\star) + \frac{1}{2}(\mathbf{z} - \mathbf{z}_a^\star)^\mathsf{T} \mathbf{\Lambda}_a (\mathbf{z} - \mathbf{z}_a^\star)$. The Gaussian integral over d dimensions is $(2\pi)^{d/2} (\det \mathbf{\Lambda}_a)^{-1/2}$, so

$$\log Z_a = -C_a(\mathbf{z}_a^\star) - \tfrac{1}{2} \log\det \mathbf{\Lambda}_a + \tfrac{d}{2} \log 2\pi + \tfrac{1}{2} \sum_k \log \frac{w_k}{2\pi},$$

which is equation (7.2), the constant being the last two terms. Two branches with the same rows and widths share d and every w_k, so the constant cancels. The log-determinant does not. With m physics rows of width σ, $\mathbf{\Lambda}_a$ is dominated by $\sigma^{-2} \mathbf{J}_{g,a}^\mathsf{T} \mathbf{J}_{g,a}$ on the solved variables. As $\sigma \to 0$, $\log\det \mathbf{\Lambda}_a$ is $-2m \log \sigma + 2 \log |\det \mathbf{J}_{x,a}|$, plus a term from the inputs' posterior, which the proof of Proposition 7.1 identifies with the predictive density. The first piece is common to the branches; the second is not whenever the branches' physics coefficients differ. So $\log Z_a$ carries $-\log |\det \mathbf{J}_{x,a}|$, which is exactly what the proposition adds back.

Exercise 7.4. The two regimes are two rows for the same closure, each with its consistency condition: throttling, $Q - c\,u\,\Delta p = 0$ with $u \le 1$, and wide open, $Q - c\,\Delta p = 0$ with $u \ge 1$. The script `ch07_valve.py` takes $c\,\Delta p = 10$ kilograms per second, a prior $u \sim \mathcal{N}(1.05, 0.1^2)$ for a valve usually driven slightly past fully open, and a meter reading $Q = 9.8$ with accuracy 0.3. Each regime is a linear-Gaussian branch. The throttling branch reads u through the meter: its posterior is 0.986 ± 0.029, inside its region, and the predictive density of the reading under it is 0.305. The wide-open branch does not read u at all, since the flow no longer depends on it. Its posterior is the prior, with mode 1.05, inside its region, and its predictive density is 1.065. Both regimes are consistent at their modes. By Proposition 7.2 each weight is the predictive density times the mass the branch's posterior puts in its own region, 0.690 for throttling and 0.692 for wide open, so the probabilities are 0.222 and 0.778; direct integration of the exact posterior gives the same to four digits. Figure 7.4 draws that posterior. It is continuous, since both rows give $Q = c\,\Delta p$ at $u = 1$, but its logarithm's slope jumps there from -17.2 just below to $+5.0$ just above. The kink is a notch between two local modes, one per regime, and a single Gaussian placed at either mode would miss the other regime entirely.

Exercise 7.5. The script `ch07_exponent.py` integrates the trunk main out in closed form, since for fixed G the objective is quadratic in h_a, and computes the exact marginal of G by a one-dimensional quadrature. With the sensor at 0.05 metres, the Laplace standard deviation of G is within 0.6 percent of the exact one at every exponent from 1 to 4, and so is that of h_a: at $p = 4$, 15.20 against 15.11 and 1.933 against 1.924. No exponent in the range makes either

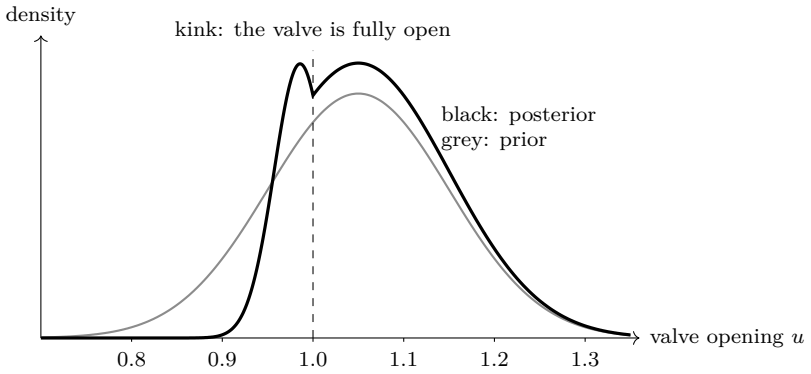


Figure 7.4: The posterior of the valve opening after a flow reading of 9.8 kilograms per second with accuracy 0.3 (black), against the prior (grey). Below $u = 1$ the valve throttles and the reading pins the opening; above it the valve is wide open and the reading says nothing about the opening. The posterior is continuous with a kink at $u = 1$, a notch between the two regimes' modes.

spread err by ten percent. What does move is the mean of G, 1.1 kg/s low at $p = 4$, a fourteenth of its spread. The exercise's premise, that a concentrated posterior would make the marginal spreads fail, is wrong here, and the reason is the third failure mode of §7.4. The precise sensor confines the posterior to a thin curved band about $h_a + (G/C)^{1/p} = 72$, and a Gaussian cannot follow a curved band. It can still match the band's two marginal spreads, which average over the curvature. Chapter 12's probes see what the spreads do not: at $p = 4$ with this sensor the objective at two spreads along the widest axis is 29 times its quadratic prediction, and only 7 percent of reweighted draws are effective. A check of marginal spreads is not a check of the Gaussian.

Exercise 7.2. With the pipe open the flow on pipe 2–3 is $m_{23} = (q_2 - q_3)/3$, so under the demand priors its prediction is -0.333 ± 0.096 with the meter's accuracy added; with the pipe shut it is exactly zero, 0.000 ± 0.020. The two predictions are separated by more than three of the wider spread, and a reading near either is decisive. A reading of zero gives $\mathbb{P}(\text{out}) = 0.990$ despite the prior of 0.05, and a reading of -0.167, halfway between the two, gives less than 10^{-4}: the out-of-service branch says the reading must be zero to within 0.02, so anything eight of its spreads away rules it out. The meter reads the quantity the outage changes most, the flow on the pipe itself. The meter on pipe 1–3 reads $m_{13} = -(q_2 + 2q_3)/3$, predicted 0.833 ± 0.150 in service and $-q_3 = 0.500 \pm 0.201$ out: the same separation and the same spreads as the meter on pipe 1–2, mirrored. At its out-branch prediction it gives $\mathbb{P}(\text{out}) = 0.315$, exactly what the pipe 1–2 meter gives at its own. It is as informative because the outage moves junction 2's demand onto pipe 1–2 and junction 3's onto pipe 1–3, by amounts that the demands' symmetric priors make equal.

Exercise 7.3. Under each branch m_{12} is a linear combination of the demands: $-(2q_2 + q_3)/3$ with both pipes in, $-q_2$ with pipe 2–3 out, $-(q_2 + q_3)$ with pipe 1–3 out, since junction 3 is then fed through junction 2, and $-q_2$ with both out, since junction 3 is islanded and its demand shed. The predictive densities at 1.50 are 0.228, 1.985, 0.297 and 1.985; the priors are 0.9025, 0.0475, 0.0475 and 0.0025; the posterior is 0.644, 0.296, 0.044 and 0.016. The meter cannot tell "pipe

2–3 out” from “both out”: in both, junction 2 is fed by pipe 1–2 alone and $m_{12} = -q_2$. Their predictive densities are identical, so their posterior ratio is their prior ratio, 19 to 1, whatever the reading. The second outage is invisible to this meter because it happens on the far side of the first; a meter on pipe 1–3 would see it at once. Pruning by weight alone would discard “both out” at a threshold of 0.02, and that is safe only because its consequence, junction 3 dark, is one an operator would see by other means.

Part III

Inference

Chapter 8

Exact inference: elimination, junction trees, Schur complements

Part II built the model and said what its posterior is. Part III is about computing it, and this chapter is about computing it exactly. The tools are the ones Chapter 5 introduced, elimination and separators, and Chapter 6 made concrete, the Schur complement. What is new is organization. Elimination can be arranged as messages passed along a tree, which gives every marginal from two sweeps; a loopy graph can be made into a tree of clusters; and, most usefully for a physical system, elimination can be organized by regions and ports, so that each region is summarized to its neighbors once and the summary is reused for every question that does not change the region. The chapter ends with two districts of a small network cut at one transfer main, each region eliminated onto the port pair of the cut, and the exactness and the reuse checked to machine precision.

8.1 What exact means

A query is exact when its answer is the marginal, the mode, or the conditional of the posterior, computed without approximation beyond floating-point arithmetic. For the Gaussian branches of Chapter 7 that means the mode from a sparse solve, the marginal variances from selected inversion, and the marginal over any subset from a Schur complement. For the discrete part it means summing over every branch, or over every branch that the graph does not let one sum separately. Both are elimination: integrate or sum out the variables one does not want, in some order, and what remains is the marginal of the ones one does. This chapter takes the arithmetic of elimination as settled and asks only how to organize it.

8.2 Messages on a tree

Take a factor graph that is a tree and pick any variable as the root. Every other node has a unique path to the root, so every node has a parent and some children. Elimination from the leaves inward is then a local operation at each node, and the local objects it passes are called *messages*.

A variable x sends to a neighboring factor f the product of the messages it has received from

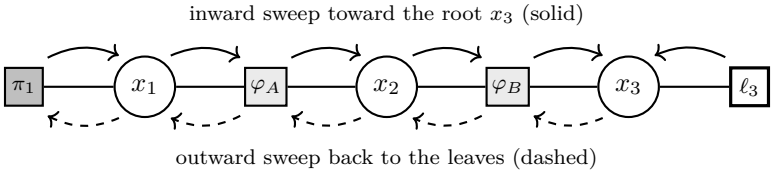


Figure 8.1: Sum-product on the three-variable chain with a prior on x_1 and a sensor on x_3. The inward sweep carries one message along each edge toward the root; the outward sweep carries one back. After both, every variable has received a message from every neighbor, and its marginal is their product (Proposition 8.1). Twelve messages, one per edge in each direction, give all three marginals.

its other neighboring factors:

$$m_{x\to f}(x) \;=\; \prod_{g\neq f} m_{g\to x}(x). \tag{8.1}$$

A factor f with scope $\{x, y_1, \dots, y_k\}$ sends to x its own function times the messages from its other variables, integrated over those variables:

$$m_{f\to x}(x) \;=\; \int \varphi_f(x, y_1, \dots, y_k) \prod_i m_{y_i\to f}(y_i)\, \mathrm{d}y_1 \cdots \mathrm{d}y_k. \tag{8.2}$$

Equation (8.2) is one elimination step: it removes $y_1, \dots, y_k$ and leaves a function of x. A leaf factor, a prior or a sensor, has no other variables and sends itself. Run the messages from the leaves to the root, and the root's marginal is the product of what arrives. Run them back from the root to the leaves, and every variable's marginal is the product of what it has received from all sides:

$$p(x) \;\propto\; \prod_{f\ni x} m_{f\to x}(x). \tag{8.3}$$

Two sweeps, every marginal, no approximation. On a tree this is exact, and it is called the *sum-product* algorithm.

Proposition 8.1 (Sum-product is exact on a tree)**.** *If the factor graph is a tree, the messages* (8.1) *and* (8.2)*, computed from the leaves inward and then back outward, give every variable's marginal by equation* (8.3)*, exactly.*

Proof. Fix a variable x. Removing x from a tree splits it into one subtree per neighboring factor f, and because the graph is a tree no factor in one subtree reads a variable of another. So the joint is $\prod_{f\ni x} \Phi_f(x, V_f)$, where Φ_f is the product of the factors in f's subtree and V_f its variables other than x, and the marginal is $p(x) \propto \prod_{f\ni x} \int \Phi_f \, \mathrm{d}V_f$. It remains to show that $\int \Phi_f \, \mathrm{d}V_f = m_{f\to x}(x)$, which is an induction on the depth of the subtree. If f is a leaf, $\Phi_f = \varphi_f(x)$ and the message is φ_f itself. Otherwise f's subtree is f together with, for each of its other variables y_i, the subtrees hanging from y_i through its other factors g; these share no variables, so $\int \Phi_f \, \mathrm{d}V_f = \int \varphi_f(x, \mathbf{y}) \prod_i \big[\prod_{g\ni y_i,\, g\neq f} \int \Phi_g \, \mathrm{d}V_g\big] \mathrm{d}\mathbf{y}$. By the induction hypothesis each inner integral is $m_{g\to y_i}$, their product is $m_{y_i\to f}$ by equation (8.1), and the outer integral is equation (8.2). The inward sweep computes these messages for the root; the outward sweep computes, for every other variable, the messages from the side of the tree that the inward sweep did not visit. □

Figure 8.1 draws the two sweeps on the chain. The count is worth noticing: two messages per edge give every marginal, where computing each marginal separately by elimination would repeat most of the work once per variable. That saving is what the organization buys, and it is the same saving that selected inversion buys against a factorization in Chapter 6.

For Gaussian factors every message is a Gaussian in one variable, hence two numbers in information form, a precision λ and an information η. Equation (8.1) adds them. Equation (8.2) is a Schur complement: form the factor's precision on its scope, add the incoming precisions to the diagonal entries of $y_1, \ldots, y_k$, and eliminate those variables. The whole computation is a sequence of small dense eliminations, one per factor, in an order fixed by the tree.

The chain of Figure 5.1 is the smallest instance. Root it at x_3. The message from φ_A to x_2 integrates x_1 out of φ_A times any prior on x_1; x_2 passes it to φ_B; φ_B integrates x_2 out and delivers a function of x_3. That is the reading of Chapter 3's worked example as a computation: the sensor's message enters at h_2, passes through the closures and balances, and arrives at G three factors away, attenuated by each.

8.3 Loops: the junction tree

Every factor graph in this book has loops, and on a loopy graph the messages (8.1)–(8.2) do not terminate and do not give exact marginals. Chapter 9 says what happens if one runs them anyway. The exact route is to make a tree.

Group the variables into overlapping *clusters* such that every factor's scope lies within some cluster, and arrange the clusters in a tree such that any variable shared by two clusters is contained in every cluster on the path between them. The second condition is the *running intersection property*, and a cluster tree with it is a *junction tree*. The intersection of two adjacent clusters is their separator. Assign each factor to a cluster containing its scope, and run the sum-product messages between clusters instead of between variables and factors: a cluster sends to its neighbor the product of its factors and its other incoming messages, integrated over everything not in the separator. Two sweeps give the marginal of every cluster, hence of every variable, exactly.

A junction tree is an elimination order in disguise, and the correspondence is the one Chapter 5 set up. Take any elimination order; each eliminated variable together with its neighbors at the time of elimination is a clique of the filled graph; the maximal cliques, arranged by the order, form a junction tree; and the largest cluster has one more variable than the order's width. The water loop in nodal form is a single cluster of two variables. The chain in row form, with its six-cycle, has a junction tree whose largest cluster is three variables, and the inputs-first order of §6.5, which created no fill, is the order that builds it. For Gaussian factors, the junction tree's messages are the block Schur complements that a sparse Cholesky factorization performs in that order; the elimination tree of the factorization is the junction tree, and Chapter 6's statement that inference is one factorization is this section's statement with the clusters made explicit.

Figure 8.2 draws the junction tree of the water loop in row-per-factor form, built from the order m_{12}, m_{13}, m_{23}, h_2, h_3 that Chapter 5 found to have width two. Eliminating m_{12} forms the clique $\{m_{12}, h_2, m_{23}\}$; eliminating m_{13} forms $\{m_{13}, h_3, m_{23}\}$; eliminating m_{23} forms $\{m_{23}, h_2, h_3\}$; the rest are contained in these. Three clusters of three variables, joined through the third by two separators of two, and each of the five rows lies in a cluster that contains its scope: the balance b_2 and the closure c_{12} in the first, b_3 and c_{13} in the second, c_{23} in the third. The variable m_{23} is in all three clusters and on every path between them, which is the running intersection property.

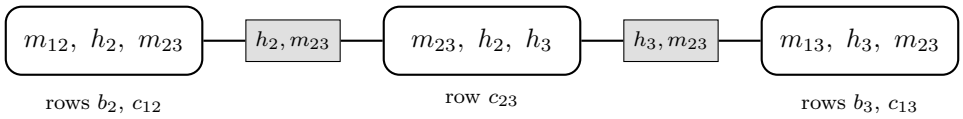


Figure 8.2: A junction tree for the water loop in row-per-factor form. Clusters (rounded) are the cliques of the elimination order $m_{12}, m_{13}, m_{23}, h_2, h_3$; separators (shaded) are their intersections. Every row is assigned to a cluster containing its scope, and the loop of the network has become a chain of clusters.

Principle 8.1. *Exact inference on a loopy graph is exact inference on a tree of clusters. The clusters are the cliques of an elimination order, the largest cluster costs what treewidth says it costs, and for Gaussian models the cluster messages are the Schur complements of a sparse factorization.*

8.4 Regions and ports: the Schur complement as a message

For a physical system there is a natural coarse junction tree: the regions of a physical partition, with the ports of each cut as separators. Proposition 5.3 says the ports do separate, so the clusters are the regions' interiors together with their ports, and the tree is the tree of regions. This section writes out what the messages are.

Partition the variables into region interiors $I_1, \dots, I_m$ and the port set S, and partition the factors accordingly: those reading only $I_r \cup S$ belong to region r, and those reading only S, such as the cut closures once one potential per cut edge has been assigned to a side, belong to the cut. The precision matrix of the whole model, ordered with the interiors first and the ports last, is then bordered block-diagonal:

$$\boldsymbol{\Lambda} \;=\; \begin{pmatrix} \boldsymbol{\Lambda}_{11} & & & \boldsymbol{\Lambda}_{1S} \\ & \boldsymbol{\Lambda}_{22} & & \boldsymbol{\Lambda}_{2S} \\ & & \ddots & \vdots \\ \boldsymbol{\Lambda}_{S1} & \boldsymbol{\Lambda}_{S2} & \cdots & \boldsymbol{\Lambda}_{SS} \end{pmatrix}, \qquad \boldsymbol{\Lambda}_{SS} = \sum_r \boldsymbol{\Lambda}_{SS}^{(r)} + \boldsymbol{\Lambda}_{SS}^{\text{cut}}, \tag{8.4}$$

with no coupling between different interiors, because no factor reads two. Figure 8.4 draws the structure for the example below.

Proposition 8.2 (The region message)**.** *Let region r's factors have precision $\boldsymbol{\Lambda}^{(r)}$ and information $\boldsymbol{\eta}^{(r)}$ on $I_r \cup S$. Their product, marginalized over I_r, is a Gaussian on S with*

$$\mathbf{S}_r \;=\; \boldsymbol{\Lambda}_{SS}^{(r)} - \boldsymbol{\Lambda}_{SI}^{(r)} (\boldsymbol{\Lambda}_{II}^{(r)})^{-1} \boldsymbol{\Lambda}_{IS}^{(r)}, \qquad \mathbf{h}_r \;=\; \boldsymbol{\eta}_S^{(r)} - \boldsymbol{\Lambda}_{SI}^{(r)} (\boldsymbol{\Lambda}_{II}^{(r)})^{-1} \boldsymbol{\eta}_I^{(r)}. \tag{8.5}$$

This is the region's message to the rest of the system, and it is the Schur complement of the region's interior in the region's own precision.

Proposition 8.3 (Exactness of the reduced system)**.** *The posterior marginal on the ports is the Gaussian with precision $\sum_r \mathbf{S}_r + \boldsymbol{\Lambda}_{SS}^{cut}$ and information $\sum_r \mathbf{h}_r + \boldsymbol{\eta}_S^{cut}$. Given the port posterior $(\boldsymbol{\mu}_S, \boldsymbol{\Sigma}_S)$, the posterior of region r's interior is Gaussian with*

$$\boldsymbol{\mu}_I = (\boldsymbol{\Lambda}_{II}^{(r)})^{-1} (\boldsymbol{\eta}_I^{(r)} - \boldsymbol{\Lambda}_{IS}^{(r)} \boldsymbol{\mu}_S), \qquad \boldsymbol{\Sigma}_I = (\boldsymbol{\Lambda}_{II}^{(r)})^{-1} + \mathbf{X}\, \boldsymbol{\Sigma}_S\, \mathbf{X}^{\mathsf{T}}, \quad \mathbf{X} = (\boldsymbol{\Lambda}_{II}^{(r)})^{-1} \boldsymbol{\Lambda}_{IS}^{(r)}. \tag{8.6}$$

Both are exact.

Proof. Proposition 8.2 is Proposition 6.3 applied to the product of region r's factors, which is a Gaussian on $I_r \cup S$. For Proposition 8.3, eliminate every interior from the whole precision at once. By the bordered structure (8.4), $\mathbf{\Lambda}_{I_r I_s} = \mathbf{0}$ for $r \neq s$, so the interior block is block diagonal, its inverse is block diagonal, and the Schur complement $\mathbf{\Lambda}_{SS} - \sum_r \mathbf{\Lambda}_{SI_r} \mathbf{\Lambda}_{I_r I_r}^{-1} \mathbf{\Lambda}_{I_r S}$ splits into one term per region. The only factors that read I_r are region r's, so $\mathbf{\Lambda}_{I_r I_r} = \mathbf{\Lambda}_{II}^{(r)}$ and $\mathbf{\Lambda}_{SI_r} = \mathbf{\Lambda}_{SI}^{(r)}$; with $\mathbf{\Lambda}_{SS} = \sum_r \mathbf{\Lambda}_{SS}^{(r)} + \mathbf{\Lambda}_{SS}^{\text{cut}}$ the Schur complement is $\sum_r \mathbf{S}_r + \mathbf{\Lambda}_{SS}^{\text{cut}}$, and the same bookkeeping gives the information vector. For the interior, the conditional of I_r given the ports involves only the factors that read I_r, and by Proposition 6.5 it is Gaussian with precision $\mathbf{\Lambda}_{II}^{(r)}$ and information $\boldsymbol{\eta}_I^{(r)} - \mathbf{\Lambda}_{IS}^{(r)} \mathbf{x}_S$. Averaging its mean over the port posterior gives the first formula of equation (8.6), and the law of total covariance, the conditional covariance plus the covariance of the conditional mean $-\mathbf{X}\mathbf{x}_S + \text{const}$, gives the second. □

Nothing is approximated. The interior covariance in equation (8.6) has two terms: the region's own conditional uncertainty given its ports, and the port uncertainty carried into the interior by the sensitivity $\mathbf{X}$ of the interior to the ports. A region whose ports are known exactly has only the first term; a region cut off from all sensors has mostly the second.

This organization has three names in three literatures. In sparse direct methods it is *nested dissection* or *substructuring*: eliminate the interiors, solve the reduced system on the separators, back-substitute. In water networks it is *skeletonization*: eliminating the interior junctions of a zone leaves an equivalent network among the boundary junctions, with equivalent pipes and equivalent demands, and every utility keeps a skeletonized model beside its full one for exactly this reason. What the utility's skeleton does not carry, and what the message does, is the uncertainty: the equivalent demand comes out with a variance. In electrical networks the same construction is *Kron reduction*, and the reduced conductance matrix is exactly the Schur complement. In graphical models it is the junction tree with regions as clusters. They are one computation, and the physical reading is the most useful: the region's message is the region as its neighbors see it, an equivalent network with error bars.

> **Principle 8.2.** *A region's Schur complement onto its ports is the region summarized to its neighbors, exactly. It is an equivalent network on the ports with uncertainty, and the rest of the system needs nothing else about the region.*

Proposition 8.4 (Messages compose)**.** *Split the interior variables into sets $I_1, \ldots, I_R$, and call the rest S. The reduced system on S obtained by eliminating every interior at once equals the one obtained by eliminating them one set at a time, in any order. If, in addition, no factor involves variables of two different interiors, it equals the sum of the regions' messages, each computed from that region's factors alone, plus the factors that involve only S.*

Proof. The reduced system is the information form of the marginal of S, by Proposition 6.3. Marginalizing a density over $I_1 \cup I_2$ at once, or over I_1 and then over I_2, gives the same marginal, since the integrals can be taken in either order; so the reduced systems agree, and by induction they agree for any number of sets in any order. When no factor involves two interiors, the joint density is a product of one group of factors per region, each group involving only that region's interior and S, times the factors on S alone. Integrating over I_r touches only region r's group, which becomes its message on S. The product of the messages and the S-only factors is the marginal, and in information form a product is a sum. □

The script `ch08_messages.py` checks both parts on the two-district network of §8.6. Eliminating district A first, district B first, or both together gives the same 2×2 port system to thirteen digits relative to its entries, and adding the two districts' separately computed messages gives it too: the transfer flow is 21.499450 kg/s by either route.

Now add one factor that joins a head of district A to a head of district B directly, without passing through the port — a pair of gauges read against each other, which is a thing a utility does when it wants to know the drop across a district rather than the level in it. Let them read $h_3 - h_5$ three gauge spreads above what the model expects. Sequential elimination is still exact. The sum of separately computed messages is not. The new factor belongs to neither district's group and touches no port, so neither message sees it: the exact posterior mean of the transfer flow is 21.5159 kg/s and the sum of messages gives 21.4994, an error of 0.016 kg/s, a tenth of the port's posterior spread. A factor across the cut that bypasses the ports makes a different partition, with a larger separator, and the districts have to be redrawn before their messages can be added.

Composition is what makes hierarchy possible. A region can itself be a union of subregions whose messages were computed separately and eliminated in any order, and Chapter 17 builds a building, a campus and a grid out of exactly this.

8.5 Reuse

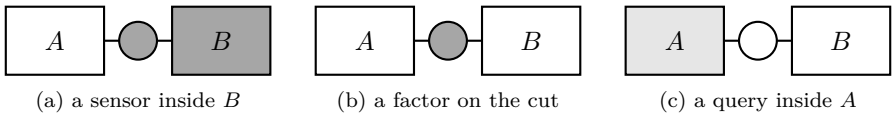
The region message depends only on the region's own factors. That single fact decides what must be recomputed when something changes, and it is the structural basis for every claim of reuse in this book.

Change	Recompute	Reuse
sensor or prior inside region r	$\mathbf{S}_r$, $\mathbf{h}_r$; port solve; back-sub in r	every other region's message
factor on the cut	port solve; back-sub where asked	every region's message
query about region r's interior	back-substitution in r	everything else
region r replaced or re-modeled	$\mathbf{S}_r$, $\mathbf{h}_r$; port solve	every other region's message

Two things in the table deserve emphasis. A sensor inside region r changes $\mathbf{h}_r$ and, in general, $\mathbf{S}_r$; but it may change neither, if what it reads is a combination of interior variables that the ports do not depend on. The worked example has such a sensor: a meter inside a region that moves the region's interior estimates and leaves its message to the rest of the system untouched, because the region's total draw through its port is fixed by its demands whatever their split. And a query about one region's interior costs a back-substitution in that region alone, which is a solve against a factorization already held. Neither the port solve nor any other region is touched.

Figure 8.3 draws the three most frequent rows of the table on the two-district network. In operation the third case dominates. Most questions are about one region's interior given everything else, and each costs a solve. The first case is the one that makes distributed estimation practical: a region's own sensors change its own message and nothing else, so a region can fold in its readings and publish a new message without any other region repeating its work.

The cost model that follows is the one Chapter 10 uses to count. Let region r have n_r interior variables and k_r ports. Computing its message costs one sparse factorization of $\mathbf{\Lambda}_{II}^{(r)}$ plus k_r solves against it, once. The port solve costs a dense factorization of a $|S| \times |S|$ matrix, where $|S| = \sum_r k_r$. A query answered by back-substitution in one region costs a solve. A change confined to one region costs that region's message again and the port solve again, and nothing



(a) a sensor inside B (b) a factor on the cut (c) a query inside A

Figure 8.3: What a change costs on the two-district network, by the table of §8.5. The circle is the port system. Dark: recomputed. Light: a back-substitution against a factorization already held. White: reused as it was. (a) A sensor inside district B changes B's message and the port solve. (b) A factor on the cut changes only the port solve. (c) A query about district A's interior is a back-substitution in A alone.

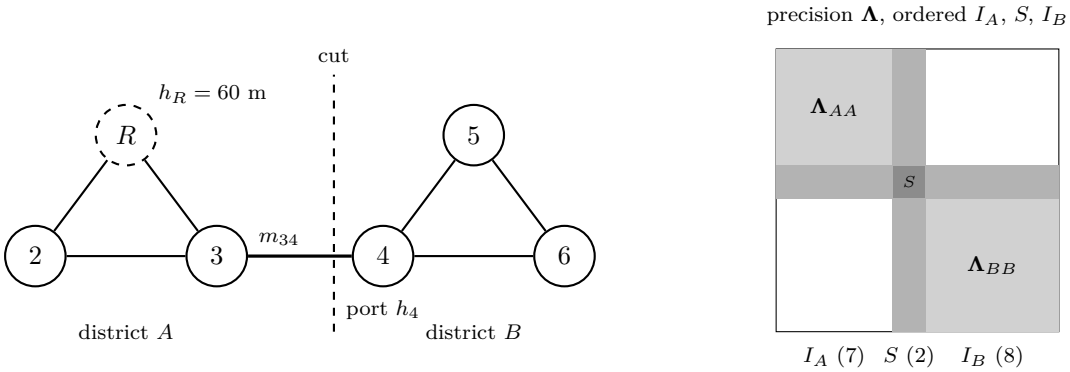


Figure 8.4: Two districts joined by one transfer main. Left: the network; the dashed line is the cut, whose port pair is the transfer flow m_{34} and the head h_4 on the B side. Right: the precision matrix with district A's seven interior variables first, the two ports in the middle, and district B's eight interior variables last. The white blocks are structurally zero: no factor reads variables from both interiors.

else. Whether this beats re-solving the whole model depends on how many regions there are, how many ports, and how many questions are asked of one model; Chapter 10 puts numbers to it, and Chapter 18 asks how to cut so that the ports are few.

8.6 Worked example: two districts, one transfer main

A reservoir R at a fixed head of 60 m feeds district A, junctions 2 and 3, through two mains and a link between them; a single transfer main, pipe 3–4, carries district B, junctions 4, 5 and 6, which are joined in a loop. Seven pipes in all, every one ductile iron with a Hazen–Williams coefficient of 130, and the five demands $d_2, \dots, d_6$ uncertain with Gaussian priors of spread 2 kg/s about 12, 8, 5, 10 and 6 kg/s. Each pipe closes with the regularized Hazen–Williams law, $\Delta h_e = K_e Q_e (Q_e^2 + q_0^2)^{0.426}$ with $Q_e = m_e/\rho$; the seven resistances run from $K = 3294$ on the largest main to $K = 57832$ on the narrowest link. Balance rows have width 0.01 kg/s and closure rows 0.01 m. Three meters read the flows on the main R–2, on the link 5–6 and on the transfer main 3–4, with accuracy 0.2 kg/s, at 14.0, -1.5 and 21.5 kg/s, values a little off the flows the nominal demands would produce. Figure 8.4 draws the network and the block structure of its precision. Every number is from the script `ch08_regions_schur.py` in the book's `scripts` directory.

The separator. In row-per-factor form the cut edge's port pair is $\{m_{34}, h_4\}$, with h_4 assigned to district B and the cut closure c_{34}, which also reads h_3, assigned to district A. The script confirms on the Markov graph of the assembled precision that no entry joins a variable of A's interior to one of B's: the 7×8 block is exactly zero. In nodal form, with the flows eliminated, the balance at junction 4 would read h_3, h_4, h_5, h_6 and d_4 together, making them a clique of five, and the smallest separator at the same cut would have three or more variables. The flow variable is worth keeping: it is the cheaper place to cut.

Exactness. The full model has 17 unknowns and 20 factors, and Gauss–Newton reaches its mode in eight iterations. Its Laplace posterior on the ports is $m_{34} = 21.4994 \pm 0.1995$ kg/s and $h_4 = 42.5404 \pm 0.4246$ m. Now eliminate each district separately. District A's factors, including the cut closure, give a 2×2 message on the ports; district B's give another; their sum is the reduced port system, and it agrees with the full model's own port message to 7×10^{-12}. Solving it gives the same two numbers to ten decimal places, and back-substituting into district A by equation (8.6) reproduces every interior marginal of the full model to ten in the mean and twelve in the spread: the demand at junction 3, for instance, is 8.1189 ± 1.8707 kg/s either way. Proposition 8.3, checked — with the caveat that "exact" here is exact for the Laplace posterior at the mode. The closures are nonlinear, and the mode has to be found before any of this can be said; what the proposition then buys is that no further approximation is made in splitting it.

The messages, read physically. District A's message has precision

$$\mathbf{S}_A \approx \begin{pmatrix} 39.45 & 11.22 \\ 11.22 & 8.73 \end{pmatrix} \quad \text{on } (m_{34}, h_4) : \tag{8.7}$$

it constrains both the transfer flow and the head at junction 4, because junction 4 is tied through the transfer main and the district-A loop to the reservoir, and because the transfer meter sits on the cut side of the split and was assigned to A. District B's message is

$$\mathbf{S}_B \approx \begin{pmatrix} 0.0835 & 0 \\ 0 & 0 \end{pmatrix}, \qquad \mathbb{E}[m_{34} \mid h_4] = 20.98 + 0 \cdot h_4, \quad \text{sd } 3.461 \text{ kg/s}. \tag{8.8}$$

District B says nothing whatever about the head at its own port: the entry is -4×10^{-12}, zero to the arithmetic. Nothing in B reaches the reservoir except the transfer main itself, so B's heads are defined only relative to h_4, and B has no opinion about where h_4 sits. What it says about the flow is that the transfer main must carry about 21 kg/s into B with a spread of 3.46 kg/s. Drop the meter on link 5–6 and the message is 21.0000 ± 3.4641 exactly: the sum of B's three demand priors, $5 + 10 + 6$, with their spreads added in quadrature, $2\sqrt{3}$. That is Proposition 1.1, exactness at a cut, delivered as a message with an error bar: the flow across the boundary equals the net demand inside, whatever the inside does, and the message is the inside's ignorance of its own total. In water the proposition is not an approximation of anything — it is mass conservation, and the head loss in the pipes of B cannot alter it. Restore the meter and the message moves to 20.9787 ± 3.4614. That 0.02 kg/s is the price of the finite width of the balance rows, which lets a reading of a difference inside B leak a little into B's total; tighten the rows and it goes away.

Had B a second connection to the supply, the message would acquire a slope in h_4, the equivalent resistance seen from junction 4, and the intercept would become an equivalent draw: the reduced network model a utility builds by hand, with the zero spread that hand-built reductions assume replaced by the spread the demand priors imply.

Reuse, two ways. Move the meter on link 5–6 from -1.5 to 0.0 kg/s. Inside B the estimates shift: d_5 from 10.39 to 8.78 kg/s and d_6 from 5.94 to 7.61, the meter redistributing demand between the two junctions. District B's message barely moves at all: its precision changes by 1.2×10^{-4} out of 0.083, and the transfer flow by 7×10^{-4} kg/s out of 21.5. The meter reads a difference of demands, and the message carries only their sum. Now instead change the prior mean of d_5 from 10.0 to 13.0 kg/s. District B's message says the transfer main carries 24.12 kg/s rather than 20.98; solving the 2×2 port system with A's old message and B's new one gives $m_{34} = 21.5099$ and $h_4 = 42.5269$, against a full re-solve's 21.5099 and 42.5270. District A's seven-variable elimination was done once and not again.

Neither reuse is exact, and the reason is worth stating because it is the one place where the nonlinearity of the closures shows. Changing anything in B moves the mode, and A's message is a Schur complement of A's rows *linearized at the mode*, so A's message moves too, by 1.5×10^{-2} here. Assemble A's message from A's own unchanged rows at the unchanged state and it is identical to machine zero: the departure is relinearization and nothing else. On a linear model the reuse would be exact; on this one it is exact to the fourth decimal and can be made exact by one more sweep. That is the honest form of the claim, and the one the later chapters rely on.

On seventeen variables none of this is worth the bookkeeping. The point is the structure: what changed inside B was paid for inside B and at the two ports, and the whole of A was reused unread. That structure is what scales.

8.7 Second worked example: two districts and a boundary main

District B of §8.6 said nothing about the head at its own port, because nothing in it reached a reservoir except the transfer main. A district with a service reservoir of its own sends a richer message. Two district metered areas sit side by side, each fed from its own service reservoir, and a boundary main joins them. Each is the network of Figure 8.5: water enters the inlet junction C from the reservoir, passes the two demand junctions E_1 and E_2, collects at the far junction H, and H is fed back to the same reservoir through the district's second feed, so the district is a ring and not a branch; a junction R is joined to both H and C by small mains. The two junctions R are joined through the boundary main, of conductance 0.06 kilograms a second per metre, and the boundary main is the cut edge. District A's reservoir stands at 42 metres and each of its demand junctions draws 20 ± 3 kilograms a second; district B's reservoir stands at 40 and its junctions draw 25 ± 4. Physics rows have width ten grammes a second. Pressure loggers of accuracy half a metre read district A's first demand junction and far junction and district B's far junction, at 23.8, 33.9 and 31.0 metres, a little off the nominal 24.6, 34.3 and 30.5. The model has thirty-one unknowns and thirty-four factors. Every number is from `ch08_districts.py`.

Head loss is taken here as a conductance times a head difference, the linearization of Chapter 1 about an operating point. That is what makes the Schur algebra of this chapter exact rather than approximate; Chapter 6 onward carries the Hazen–Williams closure itself and pays for it with an iteration.

Exactness. The port pair is the boundary flow m_w and district B's head h_{R_B}, with the boundary closure assigned to district A, which it reads through h_{R_A}. The Markov graph has no edge between the districts' interiors. The full posterior has $m_w = 0.157 \pm 0.029$ kilograms a second, a sixth of a litre a second crossing from the higher district A into B, and $h_{R_B} = 31.61 \pm 0.40$ metres. The reduced port system, the sum of the two districts' messages, gives the same two

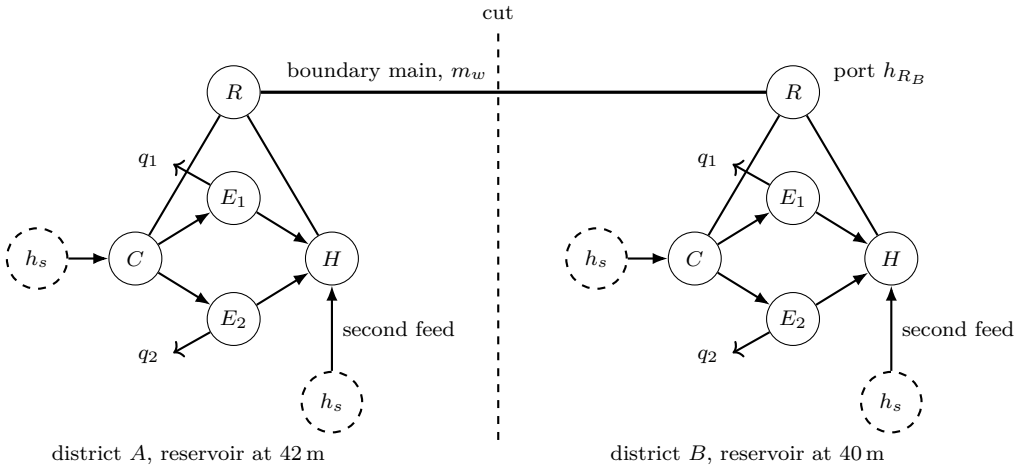


Figure 8.5: Two district metered areas joined by a boundary main. In each, water enters the inlet junction C from the service reservoir (head h_s), is drawn off at the demand junctions E_1, E_2, collects at the far junction H, which the same reservoir feeds again through the district's second feed; the junction R is joined to both. The boundary main joins the two junctions R and is the cut; its port pair is the boundary flow m_w and district B's head at R.

numbers to 1×10^{-11} and their covariance to 7×10^{-13}, and back-substitution into district B reproduces its fourteen interior marginals to 5×10^{-10}. Table 8.1 collects the numbers.

That sixth of a litre a second is 0.4 percent of district A's metered demand, and it is the quantity a district metered area exists to exclude. A boundary valve believed shut and in fact passing this much puts the district's night-flow balance out by the same amount, which is the order of a small leak.

The message is a Thevenin equivalent. District B has its own service reservoir, so its heads are defined without the boundary main, and its message constrains the port head as well as the port flow. Read as the conditional of the port head given the boundary flow, it is $\mathbb{E}[h_{R_B} \mid m_w] = 31.41 + 1.298\, m_w$ with a conditional spread of 0.43 metres: an equivalent head and an equivalent resistance, with an error bar on the first. The resistance is not quite the network's. Kron reduction of district B's conductance matrix at R, with the reservoir grounded, gives 0.698 kilograms a second per metre, a resistance of 1.433 metres per kilogram a second, and removing district B's logger from its message recovers that slope exactly, 1.4326, with the equivalent head 30.96 known only to 1.02 metres. The logger has done two things to the message. It has narrowed the equivalent head from 1.02 to 0.43 metres, and it has stiffened the apparent resistance to 1.30, because a far junction held near its reading pushes back against a change in the boundary flow more than the bare network would. The message is the equivalent network with the region's readings folded in, which is more than Kron reduction gives and different from it; Chapter 17 finds the same in a supply zone.

Reuse. Drop district A's logger reading by two metres, to 21.8. District A's demands move, from 21.04 to 23.92 and from 20.67 to 19.65 kilograms a second; district B's message does not change at all; and solving the two-variable port system with B's old message gives $m_w = 0.1396$ and $h_{R_B} = 31.592$, the full re-solve's values.

Table 8.1: The two districts: the port posterior from the full model and from the reduced port system, and district B's message read as a Thevenin equivalent, $\mathbb{E}[h_{R_B} \mid m_w] = h_{\text{eq}} + r_{\text{eq}} m_w$, with and without the pressure logger in district B, against the Kron reduction of district B's conductance matrix.

	full model	reduced port system
m_w [kg/s]	0.1571 ± 0.0288	0.1571 ± 0.0288
h_{R_B} [m]	31.614 ± 0.401	31.614 ± 0.401
district B's message	h_{eq} [m]	r_{eq} [m per kg/s]
with its logger	31.41 ± 0.43	1.298
without its logger	30.96 ± 1.02	1.4326
Kron reduction at R_B	–	1.4326

8.8 Discrete variables by region

A region with d_r discrete variables has 2^{d_r} branches, and its message is a mixture of 2^{d_r} Gaussians on its ports, each with a weight from its branch's predictive density (Proposition 7.1). Regions combine by multiplying mixtures, and the product of mixtures with c_1 and c_2 components has $c_1 c_2$; across m regions with $d = \sum_r d_r$ discrete variables in all, the port posterior has $\prod_r 2^{d_r} = 2^d$ components, which is the full enumeration again. Region structure has not reduced the count. What it has done is localize the expensive part: each region's 2^{d_r} branches are eliminated on the region's own interior, once, and only the small mixtures on the ports are multiplied. When d_r is a handful per region that is exact and affordable; when it is not, pruning components by weight and sampling the remainder are the subject of Chapter 9, and Chapter 15 applies them to pipe closures by the hundred.

The two-district network shows the count at a size small enough to list. Make three pipes shut-able, each shut with probability 0.05: pipe 2–3 in district A, and pipes 4–5 and 5–6 in district B. District A has two branches, district B four, the network eight. Each branch's weight is its prior times the predictive density of the three meters (Proposition 7.1), and Table 8.2 lists them, from `ch08_messages.py`. Before the meters the prior needs four components to reach ninety-nine percent of the weight: all pipes open, at 0.857, and each single closure, at 0.045. The prior mixture on the port head, drawn in Figure 8.6, is visibly spread: shutting pipe 2–3 moves h_4 from 43.27 to 42.78 m, and shutting both pipes at junction 5, which isolates it so that its demand is served by nothing, moves it to 53.86 — the head *rises*, because a district that has stopped drawing ten kilograms a second stops losing the head to carry them. That is the direction a power network would not go.

After the meters one component carries 0.99904 of the weight. The meter on pipe 5–6 reads -1.5 kg/s, and a branch with that pipe shut predicts a reading of zero to within the meter's accuracy; its predictive density is 10^{-11}, and every branch with pipe 5–6 out is gone. The outage of pipe 2–3 keeps 0.00096, which is the 0.001 that Chapter 9 finds for district A's mixture message on its own: the global enumeration and the region's message agree. The work divides as the text above said. District A's interior is eliminated once for each of its two branches and district B's once for each of its four, six interior eliminations, and then eight two-variable port systems are combined; the global route solves eight full models. With m regions of b branches each that is mb interior eliminations against b^m full solves, while the port combinations remain b^m unless the weights prune them, which here they do to one.

Table 8.2: The two-district network with three shut-able pipes: the eight branches, their prior probabilities, the predictive density of the three meter readings under each (relative to the largest), the posterior, and the port head h_4 in metres before and after the readings. The two branches that shut both pipes at junction 5 leave its demand unserved.

pipes shut	prior	predictive density	posterior	h_4 prior	h_4 posterior
none	0.857	1	0.97599	43.27 ± 4.62	42.5404 ± 0.4246
2–3	0.045	4.7×10^{-1}	0.02401	42.78 ± 4.90	42.0781 ± 0.6704
4–5	0.045	7.9×10^{-5}	$< 10^{-5}$	43.27 ± 4.62	42.5946 ± 0.4232
5–6	0.045	4.0×10^{-12}	$< 10^{-8}$	43.27 ± 4.62	42.5403 ± 0.4246
2–3, 4–5	0.0024	3.7×10^{-5}	$< 10^{-6}$	42.78 ± 4.90	42.1387 ± 0.6694
2–3, 5–6	0.0024	1.9×10^{-12}	$< 10^{-8}$	42.78 ± 4.90	42.0780 ± 0.6704
4–5, 5–6	0.0024	5.2×10^{-16}	$< 10^{-8}$	53.86 ± 2.31	42.6050 ± 0.4230
all three	0.0001	8.5×10^{-16}	$< 10^{-8}$	-0.300 ± 0.060	-0.5090 ± 0.0204

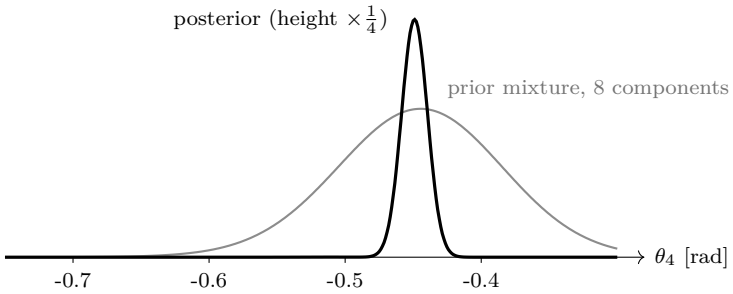


Figure 8.6: The port head h_4 of the two-district network with three shut-able pipes. Grey: the prior mixture over the eight branches, with four components carrying ninety-nine percent of the weight. Black: the posterior after the three meters, one component, drawn at a quarter of its height.

8.9 In practice

Cut at a port pair, and assign each cut closure to one side. A region's factors are those that read only its interior and its ports. The closure of a cut edge reads potentials on both sides; give it to the side that owns the near potential, and keep the far potential as the port. Getting this wrong puts a factor in both regions or in neither, and the reduced system is then wrong without any error being raised.

Check exactness once. On a representative case, solve the full model and the reduced port system and compare to machine precision, as both examples of this chapter do. A discrepancy at the eighth digit is a factor assigned to the wrong region, not rounding.

Hold the factorizations. The value of the region message is reuse, and reuse needs each region's interior factorization kept between questions. A back-substitution is then one solve against a factor already held; a change in one region is that region's elimination and the port solve.

Read the message physically. A region without a reservoir sends a message on its port flow alone, its total with an error bar. A region with a reservoir sends a Thevenin equivalent: an equivalent potential with an error bar and an equivalent resistance. If the message has neither form, a factor is missing or misassigned.

Do not replace the message by a Kron reduction. The Kron reduction of the region's conductance matrix is the message with the region's sensors and priors removed. It is right for planning with no data and wrong for estimation with data; the districts' logger changed the equivalent resistance by ten percent and the equivalent head's spread by more than half.

Keep discrete branches inside their regions. A region's branches are eliminated on its own interior, once; only the small mixtures on the ports are combined. Weight each branch by its predictive density, not by its soft-row evidence.

8.10 What is missing

Everything in this chapter was exact and everything was Gaussian per branch. Chapter 9 asks what to do when the exact route is too expensive: when the treewidth is large, when the branch count is large, or when a factor is not Gaussian and its message has no closed form. Chapter 10 then puts the exact route, the approximate route and plain sampling on one problem and counts what each costs in physics solves. And the reuse of §8.5 was stated for a change in one region; Chapter 11 treats the change that matters most in operation, one new sensor reading at a time, and shows it costs even less than a region message.

Notes and further reading

The sum-product algorithm on factor graphs, with the message equations (8.1)–(8.2) and the proof of exactness on trees, is Kschischang, Frey and Loeliger's [47]. The junction tree, the running intersection property and exact local propagation on loopy graphs are Lauritzen and Spiegelhalter's [50]; Koller and Friedman [45] give the correspondence between junction trees and elimination orders in full. For Gaussian models, Rue and Held [80] treat the same computation as sparse Cholesky, and the identification of the elimination tree with the junction tree is standard in both literatures.

Eliminating the interior of a subsystem onto its boundary is substructuring in structural mechanics and nested dissection in sparse direct methods [16, 25]; its iterative descendants are the domain decomposition methods of Toselli and Widlund [94]. In electrical networks the same Schur complement is Kron reduction [46], given its modern graph-theoretic treatment by Dörfler and Bullo [18], and the reduced network with equivalent injections is Ward's equivalent [97], still the standard external-system model in power-system analysis. Proposition 8.2 is Kron reduction of the posterior precision rather than of the conductance matrix; the book's contribution is to read it as a message on the port separator of Proposition 5.3 and to carry the priors' and sensors' information along with it, so that the equivalent has error bars. Distributed power-system state estimation by message passing between areas [14] is the closest published instance of §8.4 in a power-system setting.

In water networks, Han, Eguchi, Mehrotra and Venkatasubramanian [33] estimate a damaged network component by component, which is the elimination of each biconnected component onto

its articulation interface, and the block-Schur marginal of Proposition 8.2 is exact there whichever interface is chosen; what the interface decides is whether the interior factorizes (Chapter 5, Notes). Irofti, Romero-Ben, Stoican and Puig [35] tie the snapshots of a time window together with a persistence factor on the leak; a joint model of T network copies sharing one leak variable is the same construction, and its junction tree has the shared variable as every separator. The reuse of one factorization across many hypotheses in §8.5 is what makes a leak search affordable: a score test at the leak-free optimum costs one sparse solve per candidate junction against the cached factorization and shortlists the few hypotheses worth an exact evidence. Dellaert and Kaess [17] present the elimination algorithm, the Bayes tree and incremental re-elimination for the same sparse Gaussian computations in the language of robotics.

Summary

- On a tree, two sweeps of sum-product messages give every marginal exactly. Gaussian messages are precision-information pairs; a factor-to-variable message is a small Schur complement.
- On a loopy graph, exact inference runs on a junction tree of clusters, which is an elimination order made explicit; for Gaussian models it is the sparse Cholesky factorization.
- Physical regions with their ports as separators are a junction tree. A region's message is the Schur complement of its interior in its own precision: the region as its neighbors see it, an equivalent network with error bars. The reduced port system is exact and back-substitution recovers every interior exactly.
- The message depends only on the region's own factors. A change inside one region costs that region and the port solve; every other region is reused.
- On the two-triangle network the reduced system agrees with the full model to fourteen decimals; district B's message is its total load 2.10 ± 0.35 and nothing about the head, which is exactness at a cut with an error bar; an interior meter moves B's demands and leaves its message untouched.
- With three shut-able pipes the two-district network has eight branches; four carry ninety-nine percent of the prior and one carries 0.999 of the posterior. Regions eliminate their branches on their own interiors, mb eliminations for m regions of b branches, not b^m.
- On two districts joined by a boundary main, a district with its own service reservoir sends a Thevenin equivalent: an equivalent head with an error bar and an equivalent resistance. Without the district's logger the resistance is the Kron reduction, 1.433 metres per kilogram a second; with it, 1.30, and the equivalent head's spread falls from 1.02 to 0.43 metres.

Exercises

Exercise 8.1. Write out the sum-product messages for the chain of Figure 2.1 rooted at G, with the priors and sensor of Figure 3.1 and hard physics, and show that they reproduce the second column of Table 3.2. Which message is the Schur complement of which block?

Exercise 8.2. Construct a junction tree for the water loop in row-per-factor form (Figure 2.3) from an elimination order of width two, verify the running intersection property, and identify the separators. Then do the same with the availability a_{23} added and say which cluster holds the discrete variable.

Exercise 8.3. In the worked example, add a second transfer main from junction 6 to junction 1. The cut now has two edges and four ports. Recompute district B's message and show that its precision on the heads is no longer zero; read off the equivalent conductance between the two port junctions and compare it with the Kron reduction of the conductance matrix of district B alone.

Exercise 8.4. Prove the claim of §8.5 for the meter on m_{56}: show that, with B's only connection to the reference through the transfer main, the transfer flow is a function of the sum of B's demands alone, and hence that any interior meter whose reading is a combination of demands with zero coefficient sum leaves the message unchanged. Find an interior meter that does change it.

Exercise 8.5. Give district B two shut-able pipes, so that it has four branches, and district A one, so that it has two. Compute the eight-component mixture on the ports, its weights, and the number of components that carry more than 99 percent of the weight. At what number of shut-able pipes per region does the mixture stop being enumerable on your machine?

Solutions to selected exercises

Exercise 8.1. With hard physics and one sensor on h_2, the only row that ties the uncertain quantities to the reading is $h_2 = h_a + G/k_2$; the other rows determine h_1 and the flows, which nothing reads. The graph is then a star, one physics factor on (h_2, G, h_a) with three leaves: the prior on G, with information pair $(\lambda, \eta) = (0.0025, 0.25)$; the prior on h_a, $(0.111, 2.222)$; and the sensor on h_2, $(4, 288)$. Each leaf sends itself. The physics factor's message to G integrates h_2 and h_a against their incoming messages, $h_2 \sim \mathcal{N}(72, 0.5^2)$ and $h_a \sim \mathcal{N}(20, 3^2)$: then $G/k_2 = h_2 - h_a \sim \mathcal{N}(52, 9.25)$, so $G \sim \mathcal{N}(104, 6.083^2)$, the pair $(0.0270, 2.811)$. The marginal of G is the product of its two incoming messages, precision $0.0025 + 0.0270 = 0.0295$ and mean $(0.25 + 2.811)/0.0295 = 103.66$, a spread of 5.82. The message to h_a is $h_a = h_2 - G/2$ with $G/2 \sim \mathcal{N}(50, 10^2)$ from G's prior message, that is $\mathcal{N}(22, 10.01^2)$, and the marginal of h_a is 20.16 ± 2.87. Both are the second column of Table 3.2. The message to G is the Schur complement of the (h_2, h_a) block in the precision of the physics factor times the messages from h_2 and h_a: the two variables are eliminated onto G, which is equation (8.2) for Gaussians.

Exercise 8.3. With the second tie, region B has two port junctions, 4 and 6, and four port variables: the transfer main flows m_{34} into junction 4 and m_{61} out of junction 6, and the two heads. skeletonization of B's own conductance matrix, the three pipes 4–5, 4–6, 5–6 of conductance 10, onto junctions 4 and 6 eliminates junction 5: the 3×3 matrix $10\left(\begin{smallmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{smallmatrix}\right)$ loses its middle row and column through the Schur complement, leaving $\left(\begin{smallmatrix} 20 & -10 \\ -10 & 20 \end{smallmatrix}\right) - \frac{1}{20}\left(\begin{smallmatrix} 100 & 100 \\ 100 & 100 \end{smallmatrix}\right) = \left(\begin{smallmatrix} 15 & -15 \\ -15 & 15 \end{smallmatrix}\right)$. The equivalent conductance between the two port junctions is 15: the direct pipe of 10 in parallel with the path through junction 5, two pipes of 10 in series, which is 5. District B's message now constrains the tie flows given the port heads, and the script finds the injections into B through its ports respond to (h_4, h_6) through exactly this matrix: the message's slope is the Kron-reduced conductance, because B has no sensor and its load priors enter only the intercept. What B still cannot say is the level of its heads: the matrix has rows summing to zero, and only the transfer main to the reference through district A fixes the level. The message's precision on the heads is no longer zero, but it is singular along the direction that shifts both port heads together.

Exercise 8.4. With one tie, district B's three junctions have balances $m_{34}-m_{45}-m_{46}+q_4=0$, $m_{45}-m_{56}+q_5=0$ and $m_{46}+m_{56}+q_6=0$; their sum is $m_{34}+q_4+q_5+q_6=0$, so the transfer flow is minus the sum of the demands whatever the internal flows. The message on (m_{34}, h_4) is therefore a statement about the sum $\Sigma = q_4+q_5+q_6$ alone, and a meter changes it only if it changes the posterior of Σ given the port. A meter reading a combination $\sum_i c_i P_i$ with $\sum_i c_i = 0$ is, under priors of equal spread, uncorrelated with Σ a priori, since $\mathrm{Cov}(\sum c_i P_i, \Sigma) = \sigma^2 \sum c_i = 0$, and for Gaussians uncorrelated is independent; conditioning on it leaves Σ's distribution unchanged. The flow on pipe 5–6 is such a combination: within B, $m_{56} = (q_6 - q_5)/3$ plus a term in m_{34} that the port already fixes. The script confirms it: a meter on m_{56} changes the message by 1×10^{-11}, and a meter on $q_5 - q_6$ by 2×10^{-10}. A meter on the sum changes it completely, the transfer flow the message implies moving from 2.100 ± 0.347 to 2.299 ± 0.026; a meter on one load, q_5, changes it partly, to 2.199 ± 0.284, because one load's reading is correlated with the sum.

Exercise 8.2. The first part is Figure 8.2. In the row form of Figure 2.3, with the demands as fixed injections, the order m_{12}, m_{13}, m_{23}, h_2, h_3 has width two, and its cliques are the three clusters of the figure, with separators $\{h_2, m_{23}\}$ and $\{h_3, m_{23}\}$. The running intersection property holds: m_{23} is in all three clusters, and each head is in two adjacent ones. If the demands are uncertain variables, as in the triangle of Exercise 5.3, eliminate them first. q_2 forms the cluster $\{q_2, m_{12}, m_{23}\}$ and q_3 the cluster $\{q_3, m_{13}, m_{23}\}$, and each joins the cluster holding its flow through the separator of that flow and m_{23}. The width stays two, since the new clusters also have three variables, and the property still holds: m_{23} is now in all five clusters, and each load is in one. With the availability a_{23} added, the closure of pipe 2–3 becomes $m_{23} - a_{23}B(h_2 - h_3)$, and a_{23} joins the cluster that holds that row, the middle one, which becomes $\{m_{23}, h_2, h_3, a_{23}\}$. The discrete variable lives in one cluster, and every message out of that cluster is a two-component mixture: Chapter 7's branch structure, seen as a junction tree.

Exercise 8.5. This is the configuration of §8.8: pipe 2–3 uncertain in district A, and pipes 4–5 and 5–6 in district B, each shut with probability 0.05. Table 8.2 lists the eight components with their weights. Before the meters, four components carry more than 99 percent of the weight; after them, one does, at 0.99904. The mixture's size is the product of the regions' branch counts, so with ℓ shut-able pipes in each of m regions it has $2^{m\ell}$ components, of which the region-wise route eliminates only $m\,2^{\ell}$ interiors and combines the rest on the ports. Two regions of 20 shut-able pipes each already give $2^{40} \approx 10^{12}$ port combinations, past what a workstation enumerates in an afternoon. Beyond that, the components have to be pruned by weight and by the readings, as Chapter 9 does, and the mixture carried as its few heavy components.

Chapter 9

Approximate inference

Chapter 8 computed posteriors exactly, and said when that is affordable: when the treewidth of the factorization is small, when the number of discrete branches is small, and when every factor is Gaussian per branch. This chapter is about what to do when one of those fails. It has an unusual shape for a chapter on approximate inference, because its first and longest section is about a method that does not work on the graphs of this book. Loopy belief propagation, the standard approximate method for graphical models, fails to converge on every physical model the book has built, for a reason that is structural and worth understanding. The methods that do work are the ones that respect the structure the earlier chapters found: exact elimination within regions, approximate messages between them, and simulation inside a region that emits a compact summary to its ports. The chapter closes with the hierarchy of methods, from exact and analytical to simulation inside factors, and a rule for choosing among them.

9.1 When exact is too expensive

Three things make the exact route of Chapter 8 too expensive, and they call for different remedies.

Treewidth. The largest cluster of the junction tree, hence the largest dense block of the factorization, is too big. For the nearly-planar networks of this book this rarely happens in the Gaussian part; it happens for three-dimensional meshes and for factorizations chosen badly. The remedy is a better partition (Chapter 18) or an iterative solver in place of a direct one.

Branches. The number of discrete configurations is 2^k with k in the dozens or hundreds, as when every pipe of a network may be shut. The remedy is to prune branches by their evidence, to exploit separation between discrete variables, and to sample the rest.

Non-Gaussian factors. A factor whose message has no closed form: a bounded prior, a saturating closure that the active-set loop cannot resolve, a region whose interior physics is strongly nonlinear. The remedy is to approximate the message, either by projecting it onto a Gaussian or by representing it by samples.

What follows treats the standard approximate methods in the order a graphical-model text would, but tested on the book's models, and the test changes the conclusions.

9.2 Loopy belief propagation

The sum-product messages (8.1)–(8.2) were derived for trees. Nothing stops one from running them on a loopy graph: initialize every message, update all of them from their current neighbors, repeat, and stop when they change by less than a tolerance. This is *loopy belief propagation.* It has been extraordinarily successful in decoding and in computer vision, where the graphs are loopy, the factors are weak, and approximate marginals are what is wanted. Its properties on Gaussian models are known precisely.

Proposition 9.1 (Gaussian loopy propagation)**.** *On a Gaussian model with precision* $\mathbf{\Lambda}$*, if loopy belief propagation converges, the means it returns are exact. Convergence is not guaranteed. A sufficient condition is* walk-summability*: the spectral radius of the matrix of absolute partial correlations,* $|\mathbf{R}|$ *with* $\mathbf{R} = \mathbf{I} - \mathbf{D}^{-1/2}\mathbf{\Lambda}\mathbf{D}^{-1/2}$ *and* $\mathbf{D} = \mathrm{diag}(\mathbf{\Lambda})$*, is less than one. When it converges, the variances it returns are sums over a subset of the walks that the exact variances sum over, and are wrong in general; for models whose partial correlations are all non-negative they are underestimates.*

Proof of the claim about the means. Write the message from i to j as a precision P_{ij} and an information m_{ij}, and define for each edge the *cavity* precision and mean of i without j, $P_{i\setminus j} = \mathbf{\Lambda}_{ii} + \sum_{k\neq j} P_{ki}$ and $\mu_{i\setminus j} = (\eta_i + \sum_{k\neq j} m_{ki})/P_{i\setminus j}$. The iteration sets $P_{ij} = -\mathbf{\Lambda}_{ij}^2/P_{i\setminus j}$ and $m_{ij} = -\mathbf{\Lambda}_{ij}\,\mu_{i\setminus j}$, and the belief of i has precision $P_i = \mathbf{\Lambda}_{ii} + \sum_k P_{ki}$ and mean $\mu_i = (\eta_i + \sum_k m_{ki})/P_i$. At a fixed point, rearrange the belief of i: $\eta_i = P_i\mu_i - \sum_k m_{ki} = \mathbf{\Lambda}_{ii}\mu_i + \sum_k \big(P_{ki}\mu_i + \mathbf{\Lambda}_{ki}\mu_{k\setminus i}\big)$. So the i-th row of $\mathbf{\Lambda}\boldsymbol{\mu} = \boldsymbol{\eta}$ holds if, for every edge, $P_{ki}\mu_i + \mathbf{\Lambda}_{ki}\mu_{k\setminus i} = \mathbf{\Lambda}_{ik}\mu_k$. Write $a = \mathbf{\Lambda}_{ik}$, $p = P_{i\setminus k}$, $q = P_{k\setminus i}$, $u = \mu_{i\setminus k}$, $v = \mu_{k\setminus i}$. The beliefs are $\mu_i = (pu - av)/(p - a^2/q) = q(pu - av)/(pq - a^2)$ and, by symmetry, $\mu_k = p(qv - au)/(pq - a^2)$. Then $P_{ki}\mu_i + av = -\frac{a^2}{q}\,\frac{q(pu-av)}{pq-a^2} + av = \frac{ap(qv-au)}{pq-a^2} = a\mu_k$, which is the identity. □

The convergence condition and the characterization of the variances are Malioutov, Johnson and Willsky's walk-sum analysis, which expands the covariance as a sum over walks in the graph; the proof is longer than this book needs, and the notes give it. What the proof above shows is why a converged iteration is worth having: its means are the exact ones, whatever the loops. The condition has a physical reading. The partial correlation between two variables that share a factor measures how tightly that factor ties them given everything else. Walk-summability says the ties around every cycle must be weak enough, in a precise sense, that the influence circulating around the cycle dies out. A graph of weak, noisy pairwise factors, such as a decoding graph, satisfies it. A graph of tight physics does not.

9.2.1 On the book's models

Run Gaussian loopy propagation, in the standard pairwise form on the posterior precision, on three of the book's models: the soft flow chain with two sensors, the water loop with uncertain demands and one meter, and the two-district network of Chapter 8. Every number is from the script `ch09_approximate.py` in the book's `scripts` directory. Table 9.1 gives the result: the spectral radius that decides walk-summability, and whether the iteration converged, undamped and with damping of one half.

Not one of the physical graphs in row or nodal form is walk-summable, and on not one does the iteration converge, damped or not. Two of the three convergent cases are the two-variable reductions, which are trees, on which the messages are exact by construction.

Table 9.1: Gaussian loopy propagation on the book's models. $\rho(|\mathbf{R}|)$ is the walk-summability radius; below one guarantees convergence. Row form is the row-per-factor graph; nodal form has the flows eliminated; the last block has the demands eliminated as well. The two-district network's rows are its Hazen–Williams closures linearized at the mode, which is where its Laplace posterior lives. Physics width 0.01.

Model	Form	unknowns	$\rho(\lvert\mathbf{R}\rvert)$	converged
pumping chain	row	6	1.64	no, with or without damping
water loop	row	7	1.70	no
two-district network	row	17	2.34	no
pumping chain	nodal	4	1.80	no
water loop	nodal	4	1.50	no
two-district network	nodal	10	2.13	no
pumping chain	heads only	2	0.20	yes, exact (a tree)
water loop	heads only	2	0.18	yes, exact (a tree)
two-district network	heads only	5	0.998	yes, in 32 sweeps

The third is the interesting one, and it is a warning rather than a reprieve. The heads-only form of the two-district network — five variables joined in two loops and a transfer main — has radius 0.998. It is walk-summable, and the iteration does converge, in thirty-two sweeps to eleven digits. But it is walk-summable by two parts in a thousand. Nothing about the network was chosen to put it there; move a conductance, tighten a meter, add a junction, and the radius crosses one and the guarantee is gone with it. A method whose applicability depends on the third decimal place of a quantity nobody computes before running it is not a method one can plan around, and that, rather than the outright failures above it, is the honest case against loopy propagation on physical graphs.

Figure 9.1 follows the iteration sweep by sweep; the numbers are from the chapter's second script, `ch09_more.py`. On the water loop in row form the error of the belief means is 1.2 after the first sweep, 2.0 after the second and 1.8 after the thirtieth: the iteration neither settles nor explodes, it wanders at an error of order one hundred percent, which is worse than reporting the priors. On the chain reduced to its two heads and on the two-district network reduced to its port pair, both trees, the first sweep is exact to 10^{-11} and 10^{-14}, and nothing moves after it. The difference between the solid line and the other two is only whether the tight cycles were eliminated before the messages were passed.

Loosening the physics does not rescue it. Sweep the physics width on the water loop in row form from 0.01 to 2 kg/s, the last being physics so loose as to be nearly absent: the radius falls from 1.70 to 1.14 and never reaches one, and the iteration never converges. The reason is visible in the structure of a physics row. The partial correlation between two variables is $-\mathbf{\Lambda}_{ij}/\sqrt{\mathbf{\Lambda}_{ii}\mathbf{\Lambda}_{jj}}$, and in the row form every entry of $\mathbf{\Lambda}$ is a sum over the few rows that read both variables. A physics variable sits in two or three rows, a flow in one closure and two balances, a head in the closures of its pipes, and nothing else; only the variables that carry a prior or a sensor have anything on their diagonal that the physics did not put there. So the tie between two physics variables that share a row is of order $1/\sqrt{d_i d_j}$, with d_i, d_j the number of rows each sits in: about one half. Every variable has several such ties, the row sums of $|\mathbf{R}|$ exceed one, and so does its spectral radius. Nor does the physics width matter: scaling every physics row by



Figure 9.1: Gaussian belief propagation sweep by sweep: the largest error of the belief means against the exact posterior, on a logarithmic scale. Solid: the water loop in row form, radius 1.70; the error wanders between one and two, a hundred to two hundred percent, and never settles. Dashed: the chain reduced to its two heads, radius 0.20. Dotted: the two-district network reduced to its port pair, radius 0.29. Both reductions are trees and are exact after one sweep.

the same factor scales $\mathbf{\Lambda}_{ij}$, $\mathbf{\Lambda}_{ii}$ and $\mathbf{\Lambda}_{jj}$ together and leaves the partial correlations where they were, which is why the sweep moved the radius only as far as the priors and sensors, which did not scale, could move it. Eliminating the flows folds rows together and changes the ties a little, but a balance at a junction still ties that junction's head to each neighbor's as tightly as the conductances say, and the network's own cycles remain.

One pair shows the mechanism in numbers. The flow on pipe 1–2 and the head at junction 2 share the closure c_{12} and nothing else, and each sits in two physics rows. Without the meter their partial correlation is -0.500 at every width, which is $-1/\sqrt{2 \cdot 2}$. The meter on m_{12} adds to that variable's diagonal a weight the physics does not scale, and dilutes the tie to -0.471 at a width of 0.01 and to -0.001 at a width of ten; but the ties that no sensor touches stay at one half, and the radius of the whole never falls below 1.13.

Principle 9.1. *Loopy belief propagation is the wrong tool for a physical graph. Each physics variable is tied to a few others by factors that nothing dilutes, the partial correlations around every physical cycle are of order one half with several per variable, walk-summability fails, and the iteration does not converge. This is not a matter of tuning the widths; it is what conservation and closure do to the graph.*

The remedy is the one Chapter 8 already gave. Do not pass messages between variables around the cycles; eliminate the cycles. Group the variables into regions whose interiors contain the tight cycles, eliminate each interior exactly, and pass messages between regions. If the region graph is a tree, as it is for two regions or for a radial arrangement of regions, the region messages are exact and no iteration is needed. If the region graph has cycles, the messages between regions are loopy propagation at the region level, and the question of convergence returns, but on a graph whose factors are the region messages: dense on the ports, and, because each region's interior has absorbed its own tight ties, much weaker between regions than within them. Whether that graph

is walk-summable is a property of the partition, and Chapter 18 measures it. The reader should expect it to hold when regions are large and ports are few, and to fail when the partition is fine.

9.3 Variational methods and expectation propagation

Loopy propagation is one of a family of methods that replace an intractable posterior by the nearest member of a tractable family. The family is chosen for tractability, a Gaussian or a product of independent factors, and *nearest* is measured by a divergence.

Variational inference minimizes the divergence from the approximation to the posterior over a family that is usually a product of independent factors, one per variable or per block. It always converges, to a local optimum, and it always returns a distribution that is too narrow: the divergence it minimizes penalizes the approximation for putting mass where the posterior has none, so the approximation tucks itself inside the posterior's bulk. For a Gaussian posterior with correlated variables, a product-form approximation gets the means right and the variances too small by a factor that grows with the correlation, which for the ridges of Chapter 3 is a large factor. The Laplace approximation of §6.4 is also a member of this family, with the full Gaussian as the tractable class and the matching done at the mode rather than by a divergence; on the book's models it has served better than either, because a full Gaussian at the mode captures exactly the correlations that a product form throws away.

The size of the factor has a closed form for a Gaussian posterior.

Proposition 9.2 (Mean field on a Gaussian)**.** *Let the posterior be* $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Lambda}^{-1})$*. The product of independent Gaussians* $q = \prod_i \mathcal{N}(m_i, s_i^2)$ *that minimizes the divergence* $\mathrm{KL}(q \,\|\, p)$ *has* $m_i = \mu_i$ *and* $s_i^2 = 1/\boldsymbol{\Lambda}_{ii}$*. Since* $1/\boldsymbol{\Lambda}_{ii} \le (\boldsymbol{\Lambda}^{-1})_{ii}$*, with the ratio* $1 - R_i^2$ *where* R_i^2 *is the squared multiple correlation of variable* i *with all the others, mean field never overstates a variance and understates it by that factor.*

Proof. Up to a constant, $\mathrm{KL}(q\|p) = \frac{1}{2}\big[\sum_i \boldsymbol{\Lambda}_{ii} s_i^2 + (\mathbf{m}-\boldsymbol{\mu})^{\mathsf{T}}\boldsymbol{\Lambda}(\mathbf{m}-\boldsymbol{\mu}) - \sum_i \log s_i^2\big]$, since the expectation of $\frac{1}{2}(\mathbf{z}-\boldsymbol{\mu})^{\mathsf{T}}\boldsymbol{\Lambda}(\mathbf{z}-\boldsymbol{\mu})$ under a product of Gaussians contributes only the diagonal of $\boldsymbol{\Lambda}$ from the variances. The quadratic form is minimized at $\mathbf{m} = \boldsymbol{\mu}$; setting the derivative in s_i^2 to zero gives $\boldsymbol{\Lambda}_{ii} = 1/s_i^2$. And $1/\boldsymbol{\Lambda}_{ii}$ is the conditional variance of z_i given every other variable, by Proposition 6.5, while $(\boldsymbol{\Lambda}^{-1})_{ii}$ is its marginal variance; conditioning on correlated variables removes the fraction R_i^2 of the variance. □

On the chain after one reading the two inputs are correlated at -0.985. Mean field reports the source to ± 1.00 kg/s and the room to ± 0.49 metres, against the exact 5.82 and 2.87: the factor 0.172 on each, which Figure 9.2 draws. On all six variables of the soft chain it is far worse. Every physics variable is pinned by its rows to within the physics width once the others are fixed, so the conditional variance of each is of the order of that width, and mean field reports the source to ± 0.010 kg/s, six hundred times too narrow. Mean field on a physical graph in row form measures the physics widths and nothing else.

Expectation propagation keeps the message-passing structure of sum-product and replaces each message that has no closed form by the Gaussian that matches its first two moments, computed against the current approximation of everything else. It is the natural extension of the Gaussian machinery to the non-Gaussian factors of Chapter 7: a truncated prior, a discrete availability, a saturating closure each become a Gaussian message that is refined in place as the rest of the posterior sharpens. On a tree of regions it is exact for the Gaussian factors

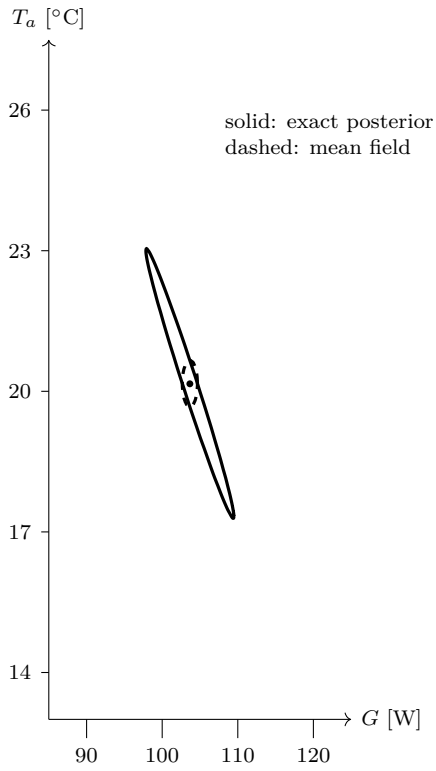


Figure 9.2: The chain's posterior over the source and the trunk main after one reading (solid) and the mean-field approximation to it (dashed), both at one standard deviation. The exact posterior is a ridge at a correlation of -0.985; mean field is the axis-aligned ellipse that fits inside it, too narrow in each coordinate by $\sqrt{1-0.985^2} = 0.172$.

and approximate only at the non-Gaussian ones, and its fixed points are often accurate where variational inference is not, because moment matching does not tuck. Its cost is that it can fail to converge, and that the moment-matched Gaussian of a bimodal message is the mixture mean and variance that §7.2.1 showed to be a value no branch believes. §9.4.1 makes that precise for a region message.

9.4 Simulation inside a region

The most useful approximate method for a physical system is not a graphical-model method at all. It is to run an ordinary physical simulation, or an ordinary Monte Carlo, inside one region, and to emit the result as a message on the region's ports.

Take a region whose interior is expensive: nonlinear closures that need a Newton solve, a dozen discrete failure states, hydrological uncertainty that enters through many parameters at once. Its message to the rest of the system, Proposition 8.2, is the marginal of its factors over its interior, a function of the ports alone. Compute it by sampling: draw the region's uncertain inputs from their priors N times; for each draw, solve the region with the ports as free boundary variables, which for a linear-Gaussian interior conditional on the draw gives a Gaussian message on the ports by the Schur complement, and for a nonlinear interior gives a Laplace message

at the solved mode. The region's message is then the equal-weight mixture of the N Gaussian components,

$$m_r(S) \;\approx\; \frac{1}{N}\sum_{i=1}^{N} \mathcal{N}\big(S;\; \boldsymbol{\mu}_S^{(i)},\; \boldsymbol{\Sigma}_S^{(i)}\big), \tag{9.1}$$

a *particle message* whose particles are Gaussians. Compress it if the rest of the system wants a Gaussian, by moment matching; keep it as a mixture if it is multimodal; keep a few components with the most weight if it is nearly so.

Three things make this attractive. The samples are drawn once per region and reused for every query the rest of the system puts to that region, because the message is a function of the ports and the queries change only what is combined with it. The expensive solves happen inside the region, on the region's own variables, with the region's own sparse structure, and the number of solves is N times the number of regions, not N times the number of global scenarios. And nothing about the region's interior needs to be Gaussian, linear or even differentiable, so long as it can be solved; the framework's assumptions apply to the message, not to what produced it. This is the sense in which the framework does not replace conventional simulation but organizes it: a region can be a legacy simulator, and its output becomes a factor. Chapter 10 counts what this costs against global sampling.

9.4.1 What a mixture message loses when collapsed

Take district A of the two-district network and make pipe 2–3 shut-able, out with prior probability 0.05. Region A's message on its ports (m_{34}, h_4) is a two-component mixture, one Gaussian per branch, weighted by the branch's evidence from the factors inside A, which include the meters on m_{12} and m_{34}, corrected for the physics determinant as Proposition 7.1 requires. The script computes both components. On the head h_4 they are -0.4490 ± 0.0096 with weight 0.999 and -0.5099 ± 0.0205 with weight 0.001: the meters inside A have all but ruled the outage out, and the two components are three standard deviations apart. Collapsing the mixture to one Gaussian by moment matching gives -0.4491 ± 0.0098, which is the dominant component with its spread inflated by two percent to cover the one-in-a-thousand chance of the other. That is a safe collapse: one component dominates, and the inflation is the honest price of the residual doubt. Had the weights been comparable, as they were at the crossover reading of §7.2.1, the collapsed Gaussian would have sat between two modes with a spread that is mostly their separation, and the rest of the system would have been told a falsehood. The rule is the one Chapter 7 stated: collapse when one component carries nearly all the weight; otherwise keep the components.

9.4.2 Worked example: a treatment works as a simulated region

Take works 1 of the two-works example of Chapter 10 as a region. Its interior holds three stream outputs and a transfer main capacity, its port is the flow it delivers to the city, S_1, and with the river drawdown fixed its message is the distribution of S_1. That message can be computed exactly on a fine grid, which is what makes the works a good test: simulate it with N draws, emit the draws as a particle message, and compare both the particles and their moment-matched Gaussian with the exact message. The city combines two such messages, one per works, and asks for the probability of a shortfall, $\mathbb{P}(S_1 + S_2 < 2{,}100)$. Every number is from the script `ch09_region.py`.

At the drought forecast drawdown of 8.5 centimetres the message is nearly Gaussian, with skewness -0.08. The Gaussian collapse gives a shortfall probability of 0.563 against the exact

Table 9.2: Works 1 as a simulated region: the city's probability of a shortfall from two particle messages of N draws each, as an rms error over 200 repetitions and as a percentage of the exact probability, and from the Gaussian collapse of the exact messages.

	drawdown 8.5 cm		drawdown 3 cm	
	rms error	% of $\mathbb{P}$	rms error	% of $\mathbb{P}$
exact $\mathbb{P}$(shortfall)	0.557	—	0.0026	—
$N = 50$	0.058	10	0.0032	123
$N = 200$	0.029	5	0.0013	49
$N = 1{,}000$	0.012	2	0.0006	23
$N = 5{,}000$	0.006	1	0.0003	10
Gaussian collapse	+0.006	1	+0.0071	272

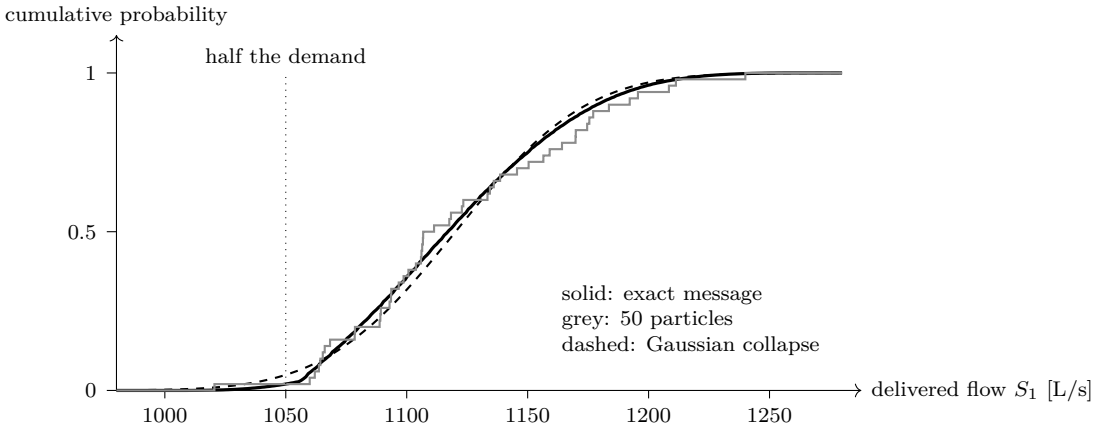


Figure 9.3: Works 1's message on its port at a drawdown of 3 centimetres, as a cumulative distribution. Solid: the exact message. Grey: a particle message of 50 draws. Dashed: the Gaussian with the same mean and spread. The exact message is skewed by the competition between the treatment and transfer limits, and its lower tail, which decides whether two works together fall below the city's demand, is thinner than the Gaussian's.

0.557, one percent off, and particle messages need about 5,000 draws each to do as well. On an ordinary day at 3 centimetres the picture reverses. The treatment streams bind in 55 percent of the draws and the main in 45, so the message is the minimum of two comparable quantities, skewed by +0.33, and the city's shortfall is a tail event, with probability 0.0026. The Gaussian collapse puts it at 0.0097, 272 percent too high: a collapsed message is right in the middle of its distribution and wrong in exactly the tail the city asks about. The particle messages converge to the tail at the rate sampling allows, with an rms error of 23 percent of the probability at 1,000 draws and 10 percent at 5,000. Table 9.2 gives the convergence at both drawdowns, and Figure 9.3 draws the 3-centimetre message.

The lesson is the one §9.4 began with, now in numbers. A region whose message is near Gaussian can be collapsed safely, and simulating it is wasted effort. A region whose message is skewed, bounded or multimodal should send its particles, or the questions downstream will be answered from the wrong tail. Which case holds can be read from the message itself, from its skewness or from a comparison of its tail with the Gaussian's, before the collapse is trusted.

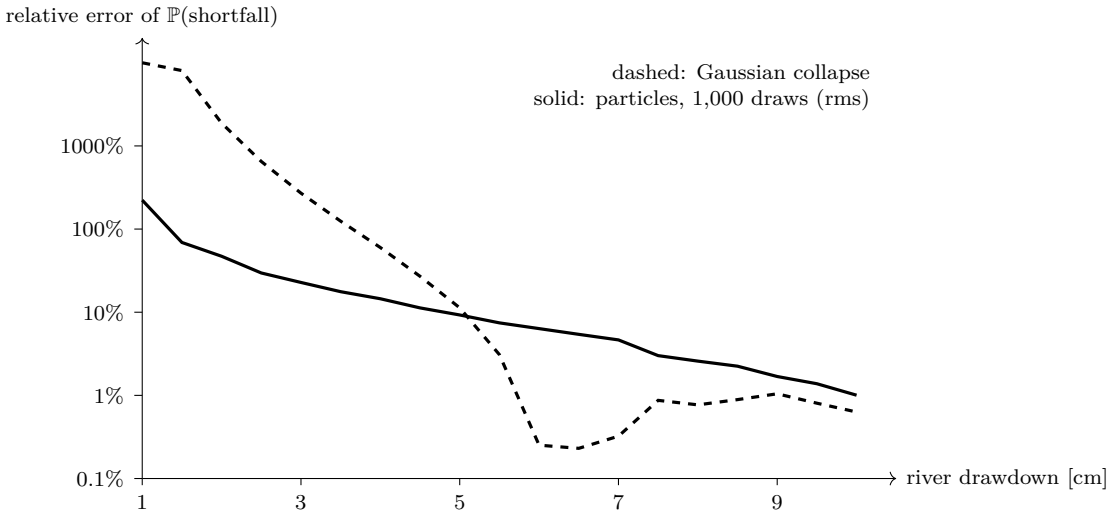


Figure 9.4: The relative error of the city's shortfall probability across drawdowns, on a logarithmic scale clipped at 10,000 percent. Dashed: the Gaussian collapse of the two works' exact messages. Solid: particle messages of 1,000 draws each, rms over 100 repetitions. The collapse is better where shortfalls are common and fails without bound where they are rare.

Figure 9.4 sweeps the drawdown. Above about 5.5 centimetres, where shortfalls are common, the collapse is within a few percent, better than 1,000 particles. Below it the shortfall becomes rare and the collapse's error grows without bound: 60 percent at 4 centimetres, 270 at 3, 1,900 at 2, and at 1 centimetre, where the exact probability is 5.5×10^{-6}, a factor of about eight hundred. Particles degrade too, since a rare event needs many draws to be seen at all; 1,000 draws give an rms error of 47 percent at 2 centimetres. But they degrade as sampling does, and more draws cure them. No number of draws cures the collapse, whose error is in the shape it assumed.

9.5 Pruning by evidence

When the branches are many, most carry negligible weight, and the weights are known before any branch is fully processed: the evidence of a branch is one number from its factorization, and a branch's prior weight bounds its posterior weight from above whenever the evidence is bounded. Discard the branches whose weight falls below a threshold and renormalize the rest.

Proposition 9.3 (Pruning error)**.** *If the discarded branches carry total posterior weight ϵ, then for every event the probability computed from the retained branches, renormalized, differs from the exact probability by at most ϵ.*

Proof. Write the exact posterior as $p = (1 - \epsilon)\, p_R + \epsilon\, p_D$, with p_R the retained branches' mixture renormalized and p_D the discarded branches' mixture renormalized. For any event E, $p(E) - p_R(E) = \epsilon\,(p_D(E) - p_R(E))$, and both probabilities lie in $[0, 1]$, so their difference is at most one in magnitude. Hence $|p(E) - p_R(E)| \le \epsilon$. □

The consequence for practice is that one can process branches in order of prior weight, accumulate the retained posterior weight, and stop when the retained fraction is high enough for the accuracy required. How far that goes by prior weight alone is a count. Table 9.3 gives it for a

Table 9.3: Pruning a hundred pipes, each out with probability 0.01, by prior weight: the prior weight discarded by keeping every branch with at most k pipes shut, and the number of branches kept.

k	discarded weight	branches kept
1	2.6×10^{-1}	101
2	7.9×10^{-2}	5 051
3	1.8×10^{-2}	166 751
4	3.4×10^{-3}	4 087 976
5	5.3×10^{-4}	7.9×10^{7}
6	7.1×10^{-5}	1.3×10^{9}
7	8.2×10^{-6}	1.7×10^{10}
8	8.4×10^{-7}	2.0×10^{11}

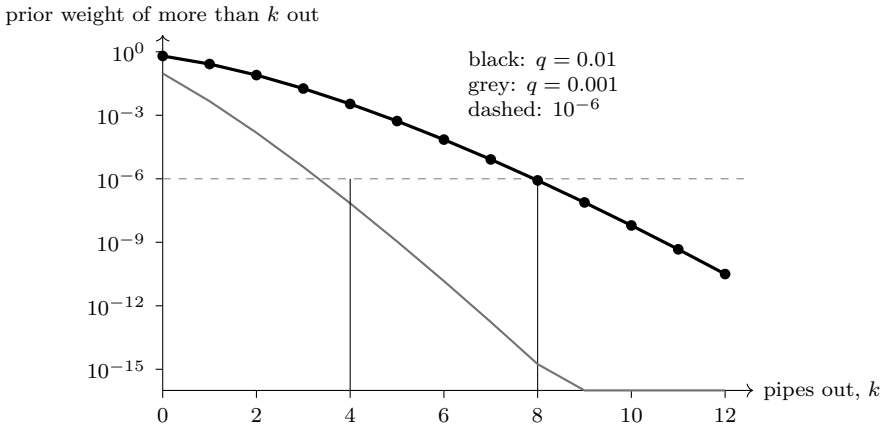


Figure 9.5: The prior weight of more than k of 100 pipes shut, each independently with probability q. Black: $q = 0.01$, which crosses 10^{-6} (dashed) at $k = 8$. Grey: $q = 0.001$, which crosses it at $k = 4$. Pruning by prior weight alone keeps every branch with k or fewer pipes out.

hundred pipes each out with probability 0.01. Keeping every branch with up to three pipes shut discards 1.8 percent of the prior weight and keeps 166 751 branches; to discard less than 10^{-6} requires every branch with up to eight pipes shut, 2×10^{11} of them, against $2^{100} \approx 1.3 \times 10^{30}$ in all. Pruning by prior weight removes nineteen orders of magnitude and still leaves two hundred billion solves. The rest must come from separation, which lets regions' branches be summed independently (Chapter 8), and from pruning by posterior weight once the readings are in, which Chapter 15 exploits.

Figure 9.5 draws the prior tail of Table 9.3 for both failure probabilities. On a logarithmic scale the tail falls faster than geometrically once k passes the mean number of failures, which is why a modest k reaches 10^{-6} at all. What the figure also shows is how much the answer depends on q: the same hundred pipes cross the tolerance at $k = 8$ or at $k = 4$, and the branch count differs by a factor of fifty thousand.

Table 9.4: The hierarchy of inference methods for a physical factor graph.

Method	When	What it costs
Exact, analytical (tree messages)	tree-structured factorization, Gaussian factors	two sweeps of small Schur complements
Exact, computational (junction tree; regions and ports)	treewidth manageable; few discrete branches per region	one sparse factorization per region, a dense port solve
Laplace per branch	rows mildly nonlinear across the posterior width	a few Gauss–Newton iterations per branch
Expectation propagation	isolated non-Gaussian unary factors on a tree of regions	repeated moment matching; may not converge
Loopy propagation between regions	region graph has cycles; region messages weak	iteration; verify walk-summability first
Simulation inside a region	interior nonlinear, discrete-heavy, or a legacy simulator	N region solves, once; particle message on the ports
Pruning by evidence	many discrete branches, weights geometric	one evidence per retained branch
Global sampling (Chapter 10)	everything else; the baseline	N solves of the whole model per query

9.6 The hierarchy, and how to choose

Table 9.4 arranges the methods of this chapter and the last from exact and analytical to simulation inside factors, with the condition under which each is the right choice.

The rule that orders the table is: eliminate exactly wherever the structure allows, approximate only at the boundaries between what has been eliminated, and never pass loopy messages through tight physics. For the models of this book the first line that applies is usually the second, regions and ports with exact interiors, with the fourth or sixth used at the few places where a factor is not Gaussian. Loopy propagation between variables is never used. Global sampling is the comparison, not the method, and the next chapter costs it.

9.7 In practice

Compute the radius before running loopy propagation. The walk-summability radius of the precision takes one eigenvalue computation. If it is above one, do not run the iteration on that graph; on a physical graph in row form it always is.

Eliminate the tight cycles first. Group variables into regions whose interiors contain the physics cycles, eliminate each interior exactly, and pass messages between regions. When the region graph is a tree the messages are exact after one sweep, as the port-pair reduction of Figure 9.1 shows.

Do not trust a converged iteration's variances. Even where loopy propagation converges its means are exact and its variances are not. Report variances from a factorization, by selected inversion.

Do not use mean field on a physical graph. A product form over physics variables reports the physics widths, six hundred times too narrow on the chain. If a cheap approximation is needed, use the Laplace posterior, which keeps the correlations.

Collapse mixtures only when one weight dominates. A two-component region message with weights 0.999 and 0.001 collapses safely; one with weights 0.7 and 0.3 does not, and the rest of the system must receive both components.

Prune with a stated error. Pruning by weight has the error bound of Proposition 9.3; state the discarded weight with every result, and expect prior weight alone to leave far too many branches for a network of a hundred uncertain components.

Collapse a message only where it is near Gaussian. Before replacing a region's particles or mixture by a Gaussian, compare the tail the downstream question needs with the Gaussian's tail. A works whose message was one percent off at one drawdown was off by a factor of nearly four at another.

9.8 What is missing

The chapter has said which approximate methods fit a physical graph and which do not, and it has counted their costs in words. Chapter 10 counts them in solves, on one problem, against enumeration and against global Monte Carlo, and states the conditions under which the factorized route wins and the conditions under which it does not. Chapter 11 then treats the approximation that operation needs most, folding in one reading at a time without re-solving, which turns out to be exact.

Notes and further reading

Loopy belief propagation on Gaussian models: Weiss and Freeman [98] proved that the means are exact at any fixed point and gave the first sufficient conditions for convergence; Malioutov, Johnson and Willsky [58] introduced walk-summability, showed that it guarantees convergence, and characterized the variances as sums over a subset of walks. The pairwise form of the iteration used in the script, as a solver for $\mathbf{\Lambda z} = \boldsymbol{\eta}$, is Shental, Siegel, Wolf, Bickson and Dolev's [84]. The observation that physics variables, sitting in only a few rows each with nothing to dilute the ties, carry partial correlations that no choice of physics width can reduce, and that physical row graphs are therefore not walk-summable, is the book's; it is elementary once stated, and it is consistent with the message-passing state estimators in the power-system literature [14] operating between areas rather than between variables.

Variational inference and its relation to belief propagation are set out by Wainwright and Jordan [96], with the divergence argument for why mean-field approximations are too narrow. Expectation propagation is Minka's [62]; Bishop [9] presents both in textbook form and MacKay [56] gives the Laplace approximation its place in the family. Particle representations of messages come from the sequential Monte Carlo literature; Robert and Casella [76] is the general reference and Särkkä and Svensson [82] the one closest to Chapter 16. The idea of a region as a black-box simulator that emits a summary to its boundary is co-simulation [26]; what §9.4 adds is that the summary is a probabilistic message on the port separator.

Summary

- Exact inference fails for large treewidth, many branches, or non-Gaussian factors; each has its own remedy.
- Gaussian loopy propagation has exact means when it converges, converges when walk-summable, and returns wrong variances. On every physical model of this book it is not walk-summable and does not converge, at any physics width, because each physics variable sits in only a few rows with nothing to dilute the ties, and the partial correlations around every cycle stay of order one half however the widths are scaled. Eliminate the cycles by regions instead.
- Variational inference is too narrow on correlated posteriors: mean field keeps the means and reports $1/\mathbf{\Lambda}_{ii}$, too narrow by $\sqrt{1 - R_i^2}$, which is 0.172 on the chain's two inputs and a factor of six hundred on its six physics variables. Expectation propagation extends Gaussian messages to non-Gaussian factors by moment matching, and collapsing a bimodal message is safe only when one component dominates.
- On the triangle in row form loopy propagation wanders at an error of one hundred percent; on tree reductions it is exact in one sweep.
- A region may be simulated and its result emitted as a particle message on its ports: solves stay inside the region, samples are reused across queries, and the interior may be anything solvable. A treatment works' message collapses to a Gaussian within one percent in a drought and misstates the city's rare shortfall by 272 percent on an ordinary day, where particles are needed.
- Pruning branches by evidence errs by at most the discarded weight. By prior weight alone, a hundred pipes at one percent still leave 2×10^{11} branches for an error of 10^{-6}.
- Order of preference: exact by regions; Laplace per branch; EP or simulation at non-Gaussian places; loopy messages only between regions and only after checking walk-summability; global sampling as the baseline.

Exercises

Exercise 9.1. Show that two variables that appear in exactly one row each, the same row, have partial correlation of magnitude one whatever the row's weight, and that a variable appearing in d rows of equal weight and unit coefficients has partial correlation $1/\sqrt{d_i d_j}$ with a neighbor it shares one row with. Compute the partial correlation between m_{12} and h_2 in the triangle row form with and without the sensor on m_{12}, and explain why the physics width drops out.

Exercise 9.2. Run the script's loopy propagation on the two-district network with the two regions replaced by their exact messages, so that the graph has only the two port variables and the cut factor. Show that it converges in one sweep and is exact, and explain why in terms of the region tree.

Exercise 9.3. Build a ring of m copies of region B from Chapter 8, each tied to the next, so that the region graph is a cycle. Compute the walk-summability radius of the port-level precision as m grows and as the transfer main's conductance varies, and find the regime in which loopy propagation between regions converges.

Exercise 9.4. Implement the particle message of equation (9.1) for district A with a shut-able pipe and uncertain conductances, using $N = 200$ draws, and compare its moment-matched

Gaussian with the exact two-branch mixture of §9.4.1. How many draws are needed before the message's mean on h_4 is within one percent of its spread?

Exercise 9.5. For L pipes each shut independently with probability q, find the number of pipes shut beyond which the discarded weight is below 10^{-6} for $L = 100$, $q = 0.01$, and count the retained branches. Repeat for $q = 0.001$, and say what the comparison implies for pruning by prior weight alone.

Solutions to selected exercises

Exercise 9.1. If variables i and j appear only in one row, with coefficients h_i, h_j and weight w, their block of $\mathbf{\Lambda}$ is $w\left(\begin{smallmatrix} h_i^2 & h_i h_j \\ h_i h_j & h_j^2 \end{smallmatrix}\right)$, and their partial correlation is $-wh_ih_j/\sqrt{wh_i^2 \cdot wh_j^2} = \mp 1$: magnitude one, whatever w. If i sits in d_i rows and j in d_j, all with unit coefficients and weight w, and they share one row, then $\mathbf{\Lambda}_{ii} = d_i w$, $\mathbf{\Lambda}_{jj} = d_j w$ and $\mathbf{\Lambda}_{ij} = w$, and the partial correlation is $-1/\sqrt{d_i d_j}$. In the triangle, m_{12} sits in b_2 and c_{12} with unit coefficients, and h_2 in c_{12} and c_{23} with coefficient B: $\mathbf{\Lambda}_{m_{12}m_{12}} = 2w$, $\mathbf{\Lambda}_{h_2h_2} = 2B^2w$, $\mathbf{\Lambda}_{m_{12}h_2} = Bw$, and the partial correlation is $-Bw/\sqrt{2w \cdot 2B^2w} = -\frac{1}{2}$, independent of w and of B, which is the script's -0.500 at every width. The width drops out because every entry scales with w. With the meter of weight w_m on m_{12}, $\mathbf{\Lambda}_{m_{12}m_{12}} = 2w + w_m$ and the partial correlation is $-1/\sqrt{2(2 + w_m/w)}$; at a physics width of 0.01 and a meter of 0.02, $w_m/w = 0.25$ and the value is $-1/\sqrt{4.5} = -0.471$. Now the width matters, because the meter's weight does not scale with it; but only the ties at metered variables are diluted.

Exercise 9.5. The number of pipes shut is binomial with $L = 100$ and $q = 0.01$, so the prior weight of having more than k out is the binomial tail, and the branches kept number $\sum_{j \le k} \binom{100}{j}$. Table 9.3 gives both: the tail first falls below 10^{-6} at $k = 8$, with 8.4×10^{-7} discarded and 2.0×10^{11} branches kept. At $q = 0.001$ the mean number out is 0.1 instead of 1, and the tail falls below 10^{-6} already at $k = 4$, with about 4.1×10^{6} branches kept: fifty thousand times fewer, for components ten times more reliable. Pruning by prior weight works when the expected number of simultaneous failures is well below one, and fails as it approaches one; a large network of ordinary components is in the second regime, which is why the count must be cut further by separation and by the readings.

Exercise 9.2. With each region replaced by its exact message, Proposition 8.2, the graph left to the iteration has two port variables and the cut factor between them, with each region's message attached to its port as a factor on that port alone. That graph is a tree, and on a tree Gaussian propagation is sum-product, exact after one sweep from the leaves inward and back, by Proposition 8.1. The dotted curve of Figure 9.1 is this run: its error is 10^{-14} after the first sweep and nothing moves afterward. The walk-summability radius of the port-level precision is 0.29, well inside the unit circle, but it is not needed: exactness on a tree does not depend on convergence of an infinite walk sum. The region tree has done what the loopy iteration could not, by eliminating every cycle inside the regions before any message was passed.

Exercise 9.3. The script `ch09_region.py` builds the ring in nodal form: each copy of district B has junctions 4, 5 and 6 and its three pipes, junction 6 of each copy is joined to junction 4 of the next, junction 4 of the first copy is the reservoir, and every junction carries a demand

prior of spread 0.2. Eliminating each region's interior junction leaves the port-level precision on the heads at junctions 4 and 6. On the port variables one at a time, loopy propagation never converges: the walk-summability radius is between 1.07 and 1.83 for every ring from $m = 3$ to $m = 20$ and every transfer conductance from 1 to 30, and the iteration diverges in every case. A region's two ports are tied tightly through its own pipes, and that alone breaks the scalar iteration. Propagation between regions should pass one message per region, over both its ports at once. Done that way it converges, and is exact, for the three-region ring at every conductance, for four regions up to 10, and for six only at a tie of 1; from ten regions on it diverges at every tie. The reason is the region graph. In nodal form the load priors couple each angle to its neighbours' neighbours, so after the interiors are eliminated each region is joined to its second neighbours as well as its first. Three regions make a single loop, on which Gaussian propagation's means converge. From six regions on, every region has four neighbours, and the ring has become a ring of chords. The block walk-summability radius, a sufficient test, is below one only for the three-region ring with the weakest tie, 0.97. It certifies none of the other cases that converge, and it correctly refuses all the ones that do not. The regime in which propagation between regions converges is a coarse partition of a nearly tree-like region graph, which is the advice of §9.2 read backwards. A ring should be cut open, by merging two regions across one tie so that the region graph becomes a tree, or treated by a junction tree over regions, as Chapter 18 does.

Chapter 10

Enumeration, Monte Carlo, and factorized inference

Chapter 5 counted the cost of factorized inference in operations and warned that the count is not an argument against sampling. This chapter makes the comparison properly. It takes one small physical problem with a closed-form answer, computes the same two quantities by enumeration, by Monte Carlo, and by a factorized computation that exploits a separator of the model, and costs each in the one currency that matters for a physical system: the number of times the physics has to be solved. The result is not that sampling loses. It is that the two routes are good at different things, that the factorized route's advantage is reuse, and that reuse is worth a great deal when many questions are asked of one model. A second example, two treatment works that share a river, repeats the comparison on nine inputs and shows what a shared cause does to the cut. The chapter ends with the claim the book is prepared to defend, in three strengths, and with what has been established for each.

10.1 The question, stated honestly

A Monte Carlo method draws the uncertain inputs from their priors, solves the physics for each draw, and averages. It answers every question this book has posed. It computes expectations, tails, conditionals by filtering the draws, counterfactuals by re-drawing under a changed prior, and attributions by any of the standard variance decompositions. It does not care how many uncertain inputs there are: a hundred thousand draws from a space of 2^{60} states cost a hundred thousand solves. Nothing in Part II gives the factorized route an advantage on those grounds, and a claim that it "handles combinatorial uncertainty without enumeration" is a claim that sampling already makes good on.

What sampling pays for is the solve. Each draw is one evaluation of the physics, and for a real network that evaluation is a Newton iteration on thousands of unknowns, or an optimal power flow, at a cost of a fraction of a second to many seconds. A hundred thousand draws is hours. The question is therefore not which method can answer, but how many solves each needs for a given accuracy, and, since a model is rarely asked one question, how many solves the second and third and tenth questions need once the first has been answered. That is the comparison this chapter runs.

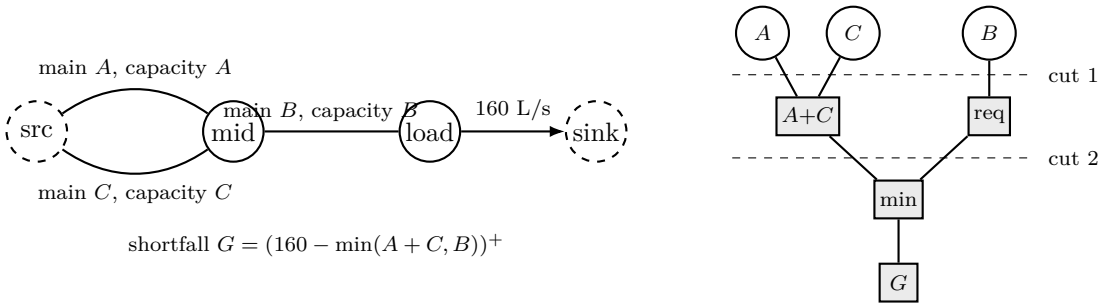


Figure 10.1: Left: the three-main scheme. Two mains in parallel feed the break-pressure tank; one main feeds the service reservoir. Right: the dependency graph of the compiled model, inputs at the top and the shortfall at the bottom. Cut 1 separates the three capacities from everything below; cut 2 separates (A+C, req), two quantities, from the shortfall. Each cut is a separator in the sense of Chapter 5, and §10.6 counts what each saves.

10.2 The problem

Two raw-water mains, A and C, run in parallel from a source to a break-pressure tank; a third main, B, runs from the tank to a service reservoir that must be filled at 160 litres a second. Each main has a capacity, the most it may carry, set by its pressure capacity and its pumps. The capacities are uncertain and independent: A and C uniform on $[70, 120]$ litres a second, B uniform on $[140, 180]$. What can be delivered is the smaller of what the parallel pair can carry, $A + C$, and what main B can carry; whatever the reservoir needs beyond that is unserved. The shortfall is

$$G \;=\; \big(160 - \min(A + C,\; B)\big)^{+}, \tag{10.1}$$

and because it has this closed form its expectation, variance and probability of being zero are known exactly, by the derivation in the solution to Exercise 10.1: $\mathbb{E}[G] = 16/3$ litres a second, $\mathrm{Var}[G] = 41.956$, $\mathbb{P}(G = 0) = 0.46$. That is the whole reason this scheme was chosen: on a real network each evaluation of G is a hydraulic solve and the truth is unknown, so no method can be scored; here every method below is scored against the answer. The Shapley shares of the variance, a standard attribution of $\mathrm{Var}[G]$ to the three capacities that Chapter 14 defines, have no such short closed form. Their reference values are computed with the closed-form G on a grid of 241 points per capacity, far finer than any grid scored below; on that grid the same sums return $\mathbb{E}[G]$ within 3×10^{-5} of $16/3$. Main B accounts for 93.60 percent and A and C for 3.20 each. Every method is scored against these values.

Figure 10.1 draws the network and the dependency graph of the model as compiled, which is the factor graph of the closures with the balance rows absorbed. Every number of the three-main comparison is computed by the script `ch10_three_way.py` in the book's `scripts` directory, which needs no engine and no results file: the shortfall has a closed form, so the counts below are counts of evaluations of it, which is what a hydraulic solve would cost on a network where it has none. The second example of §10.7 is computed by `ch10_more.py`.

10.3 Enumeration: the product grid

Discretize each capacity on n points and take the product: n^3 combinations, each with a known probability, each solved once. Expectation is the weighted sum of the n^3 shortfalls; so is any other

Table 10.1: Product-grid quadrature on the three-main problem: grid size, solves, and the error of the mean, the variance, and main B's Shapley share against the closed form. The share costs no solves beyond the grid.

n	solves	$\mathbb{E}[G]$ error [L/s]	$\mathrm{Var}[G]$ error	share B [%]	share error [pp]
9	729	0.045	2.47	92.47	1.12
17	4,913	0.012	0.60	93.30	0.29
25	15,625	0.0054	0.27	93.47	0.13
33	35,937	0.0026	0.16	93.52	0.075
41	68,921	0.0012	0.11	93.55	0.048

functional. This is enumeration, called product-grid quadrature when the inputs are continuous, and it is the exact marginalization of the discretized model. Table 10.1 gives its convergence. At $n = 9$, which is 729 solves, the mean is within 0.045 litres a second of the closed form; at $n = 41$, 68,921 solves, within 0.0012. The error falls roughly as n^{-2}, as quadrature of a piecewise-smooth integrand should, and the cost rises as n^3.

That the error falls as n^{-2} deserves a proof, because the familiar error bound of the trapezoid rule assumes a smooth integrand and G is not smooth: it bends where $A + C = B$, where $A + C = 160$ and where $B = 160$. The study's grid is the trapezoid rule on n evenly spaced points per capacity, with step h and half weights at the two ends.

Proposition 10.1 (Trapezoid rule across kinks)**.** *Let f be continuous on $[a, b]$ and twice differentiable with $|f''| \le M$ except at K points, at each of which its slope jumps by at most J. The trapezoid rule with step h has error at most $(b-a)Mh^2/12 + KJh^2/8$. On a product grid in d coordinates, for a function whose one-dimensional sections all obey these bounds, the error is at most d times the one-dimensional bound.*

Proof. On a cell $[x_0, x_0+h]$ with no kink, the rule integrates the chord through the two end values, and f differs from its chord by $\frac{1}{2}f''(\xi)(x - x_0)(x - x_0 - h)$ for some ξ in the cell. Integrating over the cell gives an error of $h^3|f''(\xi)|/12$ at most, and the $(b-a)/h$ cells contribute at most $(b-a)Mh^2/12$. On a cell with a kink at $x_0 + s$, write $f = \ell + k$, where ℓ continues the left-hand piece smoothly across the kink and $k(x) = j\,(x - x_0 - s)^+$ carries the jump j in slope. The part ℓ is covered by the first bound. The part k has integral $j(h-s)^2/2$ over the cell and trapezoid value $\frac{h}{2}\,j(h-s)$, and their difference is $\frac{1}{2}js(h-s)$, at most $jh^2/8$, reached at $s = h/2$. There are K such cells. On a product grid the rule is the one-dimensional rule applied coordinate by coordinate, $Q_1 \cdots Q_d$ in place of $I_1 \cdots I_d$, and the difference telescopes: $Q_1 \cdots Q_d - I_1 \cdots I_d = \sum_k Q_1 \cdots Q_{k-1}(Q_k - I_k)I_{k+1} \cdots I_d$. Term k is the one-dimensional error for a function of x_k alone, the section integrated over the later coordinates, which has no more kinks or curvature than the sections themselves, averaged with the positive weights of the earlier coordinates, which sum to one. □

The shortfall is piecewise linear, so $M = 0$ and only the kink term remains: the error is of order h^2 with a constant set by the jumps in slope. The ratio of the error to h^2 bears this out. It is 0.0012, 0.0013 and 0.0008 per (L/s)^2 at $n = 9$, 17 and 41. It does not settle to one constant, because the kinks fall at different places in their cells as n changes, and the $s(h-s)$ of the proof moves with them.

Table 10.2: Monte Carlo on the three-main problem. Top: the mean, rms error over eight seeds. Bottom: main B's Shapley share by pick-freeze, which needs six further sample sets, $7N$ solves in all, rms over eight seeds.

N	solves	rms error of $\mathbb{E}[G]$ [L/s]	rms error of $\mathrm{Var}[G]$
300	300	0.46	3.42
1,000	1,000	0.15	0.73
3,000	3,000	0.13	0.84
10,000	10,000	0.064	0.26
N	solves	share B [%]	rms error [pp]
300	2,100	92.7	2.67
1,000	7,000	92.5	2.22
3,000	21,000	92.4	1.60

10.4 Monte Carlo

Draw N triples of capacities, solve each, average. Table 10.2 gives the root-mean-square error of the mean over eight independent seeds: 0.46 litres a second at 300 draws, 0.064 at 10,000. The error falls as $N^{-1/2}$, as it must.

Proposition 10.2 (Monte Carlo error)**.** *Let $G_1, \dots, G_N$ be independent draws of a quantity with mean μ and variance σ^2. The sample mean $\bar{G}_N$ has expectation μ and root-mean-square error $\sigma/\sqrt{N}$, whatever the number of uncertain inputs behind each draw.*

Proof. $\mathbb{E}[\bar{G}_N] = \frac{1}{N}\sum_i \mathbb{E}[G_i] = \mu$. The variance of a sum of independent terms is the sum of their variances, so $\mathrm{Var}[\bar{G}_N] = N\sigma^2/N^2 = \sigma^2/N$, and the rms error of an unbiased estimator is the square root of its variance. Nothing in the argument refers to the inputs. □

The constant is the standard deviation of the quantity itself, $\sigma = \sqrt{41.956} = 6.48$ litres a second. The proposition predicts 0.374, 0.205, 0.118 and 0.065 at the four sample sizes of Table 10.2. The script measured 0.46, 0.15, 0.13 and 0.064, each an rms over eight seeds. An rms over eight seeds is itself uncertain by about a quarter, $1/\sqrt{2 \cdot 8}$, so the agreement is as close as eight seeds allow.

Set the two side by side on the mean. The grid at 729 solves has a smaller error than Monte Carlo at 10,000, and Proposition 10.2 says how many draws would match it: $(6.48/0.045)^2$, about 20,600, a factor of twenty-eight in solves. The two figures bound the same gap: at least fourteen as measured, since 729 grid solves beat 10,000 draws, and about twenty-eight by the $\sigma/\sqrt{N}$ law. At $n = 17$ the grid's error of 0.012 would take about 275,000 draws, fifty-six times its 4,913 solves. This is real, and it is not a property of graphical models. It is the usual advantage of quadrature over sampling in low dimension. In solves the grid's error falls as $(n^d)^{-2/d}$, the $-2/d$ power of the solves, against sampling's $-1/2$; the two exponents cross at $d = 4$, and beyond four inputs the grid loses for all but the smallest budgets. The book does not claim this factor as its own.

10.5 Attribution: where the structural difference appears

Now ask the second question: how much of the variance of G is due to each capacity? Chapter 14 defines the Shapley share; for now it is enough that computing it needs conditional expectations of G given each subset of the inputs, $\mathbb{E}[G \mid A]$, $\mathbb{E}[G \mid A, B]$, and so on, for every subset.

On the grid, every such conditional expectation is a partial sum over solves already made: fix A at a grid value, sum over the B and C grid values with their weights. The joint distribution is held on a product set, and marginalizing over some coordinates is summing over them. The attribution costs no solves beyond the grid, and at $n = 17$, 4,913 solves, main B's share is within 0.29 percentage points of the reference.

Proposition 10.3 (Conditional expectations on a product grid)**.** *Let the inputs be independent and discretized on a product grid $x = (x_1, \dots, x_d)$ with weights $w(x) = \prod_i w_i(x_i)$, and let G be known at every grid point. For any subset u of the inputs, the conditional expectation of G given $X_u = x_u$ in the discretized model is*

$$\mathbb{E}[G \mid X_u = x_u] \;=\; \sum_{x_{-u}} \Big(\prod_{i \notin u} w_i(x_i) \Big)\, G(x_u, x_{-u}), \tag{10.2}$$

a partial sum over values already solved, and $\mathrm{Var}(\mathbb{E}[G \mid X_u])$ for all 2^d subsets needs no further solve.

Proof. The conditional probability of x_{-u} given x_u is $w(x) / \sum_{x'_{-u}} w(x_u, x'_{-u})$. The denominator is $\prod_{i \in u} w_i(x_i) \prod_{i \notin u} \sum_{x_i} w_i(x_i) = \prod_{i \in u} w_i(x_i)$, since each marginal sums to one, so the ratio is $\prod_{i \notin u} w_i(x_i)$. The variance of the conditional expectation is the sum over x_u of $\prod_{i \in u} w_i(x_i)$ times the square of (10.2), less the square of the mean. Every term is a value of G at a grid point. □

The proposition leans on the product form twice: the inputs are independent, and the grid is a product set. A cloud of random draws lacks the second property. No two draws share a value of A, so there is nothing to sum over with A held fixed, and a conditional expectation has to be manufactured by pairing draws, which is what pick-freeze does.

By sampling, a conditional expectation given a subset needs a sample designed for it. The standard construction, pick-freeze, draws a second independent sample and recombines the two so that some inputs are held fixed across a pair of draws; for three inputs it needs six further sample sets, $7N$ solves in all. At $N = 3{,}000$, which is 21,000 solves, main B's share is 2.64 points off, and the error is not falling smoothly with N because the estimator's variance is large. To match the grid's 0.29 points would take, on the $N^{-1/2}$ law, some eighty times the samples.

Figure 10.2 draws all four curves on one plot of error against solves. The two mean curves are the quadrature-versus-sampling story: parallel on the log-log plot, offset by that factor. The two attribution curves are the structural story: the grid's attribution error is its mean error's twin, at the same solves, because it costs no more; the sampling attribution sits an order of magnitude above the sampling mean, because every conditional expectation is a new sample.

> **Principle 10.1.** *The factorized route does not win by avoiding a large state space. It wins when a computation held on a product set answers the second question from the solves that answered the first. Its advantage is reuse, and it is measured in solves per question, not in solves per answer.*

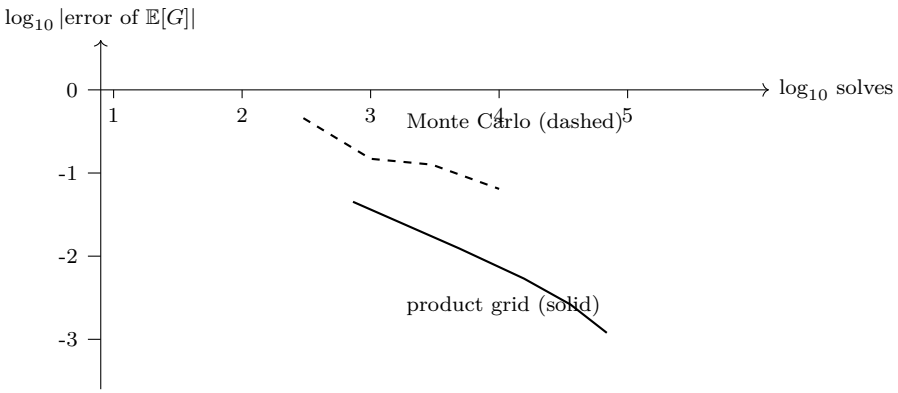


Figure 10.2: Error against solves on the three-main problem, from the committed study results. Solid: product grid; dashed: Monte Carlo. Black: error of the mean in litres a second; grey: error of main B's Shapley share in percentage points. The grid's attribution rides on its mean because it costs no further solves; the sampling attribution needs $7N$ solves and its error is an order of magnitude above the sampling mean's.

10.6 The factorized route: cuts of the dependency graph

The grid of §10.3 enumerated n^3 combinations of the capacities. It did not have to. The dependency graph of Figure 10.1 shows that the shortfall depends on the capacities only through two intermediate quantities, $A + C$ and the requirement at the tank, and on those only through their minimum. Each level of the graph is a separator in the sense of Chapter 5: given the quantities on a cut, what lies below is independent of what lies above.

Enumerate on a cut instead of on the inputs. The pair $(A + C,\ B)$ takes $(2n - 1)\,n$ distinct values on the grid, not n^3, because $A + C$ takes $2n - 1$ distinct values and each carries a known probability from the convolution of A's and C's. Solving the physics once per distinct value of the cut and weighting by the cut's distribution gives the same expectation as the full grid, exactly, because the expectation of a function of the cut is the same whether one sums over the inputs or over the cut with its induced distribution. Table 10.3 gives the ladder of cuts and the distinct solves each needs. At $n = 17$ the cut $(A + C,\ B)$ needs 561 solves against the grid's 4,913, and the script verifies that the two means agree exactly in floating point, and at $n = 41$ to 10^{-14} litres a second. The cut below it, the single quantity that G depends on, needs fewer than n, 33 at $n = 41$: the whole problem is one-dimensional at that level, and G's distribution is the pushforward of a one-dimensional distribution through a scalar function.

Proposition 10.4 (Enumeration on a cut)**.** *Let X take values on a finite grid with probabilities $p(x)$, and let the quantity of interest depend on X only through a cut $Z = c(X)$, so that $G = h(Z)$. Then for any function ϕ*

$$\mathbb{E}[\phi(G)] \;=\; \sum_{z} \phi\big(h(z)\big)\, q(z), \qquad q(z) = \sum_{x:\, c(x)=z} p(x), \tag{10.3}$$

which needs one evaluation of h per distinct value of the cut on the grid. If a component of the cut is a sum of independent inputs, its distribution is the convolution of theirs.

Proof. Group the terms of $\sum_x \phi(h(c(x)))\, p(x)$ by the value of $c(x)$. Within a group the factor

Table 10.3: The cut ladder of the three-main model: for each separator of the dependency graph, its dimension and the number of distinct physics solves needed to compute $\mathbb{E}[G]$ exactly on an n-point grid. The bottom row is the full product grid.

cut (frontier)	dim	$n = 9$	$n = 17$	$n = 25$	$n = 33$	$n = 41$
$\{f_B\}$: what G reads	1	8	14	20	26	33
$\{A{+}C,\ \text{req}\}$	2	85	297	637	1,105	1,701
$\{A{+}C,\ B\}$	2	153	561	1,225	2,145	3,321
$\{A,\ C,\ \text{req}\}$	3	405	2,601	8,125	18,513	35,301
$\{A,\ B,\ C\}$: the product grid	3	729	4,913	15,625	35,937	68,921

$\phi(h(z))$ is common and the probabilities add to $q(z)$. For a component $Z_1 = X_1 + X_2$ with X_1 and X_2 independent, $q_1(z_1) = \sum_{x_1} p_1(x_1)\, p_2(z_1 - x_1)$, which is the convolution. □

On the three-main grid with trapezoid weights, $n = 5$ puts weights $(2, 4, 4, 4, 2)/16$ on the capacities of A and of C. The convolution puts $(4, 16, 32, 48, 56, 48, 32, 16, 4)/256$ on the nine values of $A + C$, a discrete triangle, which is what the sum of two uniforms is. Recomputing the grid means through the cut reproduces the full-grid means to 2×10^{-15} at every n of Table 10.1. The evaluations of h are the physics. The convolution is arithmetic on weights and costs no solve, because in this model two parallel mains of capacities A and C act on the rest of the scheme exactly as one main of capacity $A + C$ does, which is what equation (10.1) says.

This is Chapter 5's count made physical. The exponent of the cost is the dimension of the cut, not the number of inputs; the cut is a separator of the model's own graph; and the model exposes it, because the compiled dependency graph is the factor graph with the balances folded in. Nothing was approximated, and every functional of G, not just its mean, comes from the same 561 solves, since they determine G's distribution.

Two honest qualifications. A sampler can exploit the same cut: draw $A + C$ from its convolution instead of drawing A and C, and the draws of the two inputs collapse to one. The error still falls as $N^{-1/2}$, so the cut helps the sampler less than it helps the grid, but it helps. And a cut of dimension two on a three-input problem is a modest saving; the ladder matters because on a large model the cuts are the ports of Chapter 8, their dimension is the port count, and the input count is in the hundreds.

10.7 Second worked example: two works on one river

A city draws 2,100 litres a second, supplied by two treatment works. Each works has three treatment streams in parallel, each rated between 350 and 450 litres a second, uniform and independent. Both works abstract from the same river, and as the river falls the intakes draw against a greater lift and foul faster, so every stream's output drops — here by 1.5 percent per centimetre of drawdown. The drawdown D is uncertain, uniform on $[0, 10]$ centimetres, and it is the *same number* for both works, because there is only one river. Each works delivers through its own transfer main, whose capacity is uncertain, uniform between 1,056 and 1,257 litres a second. A works delivers the smaller of its derated treatment output and its main's capacity,

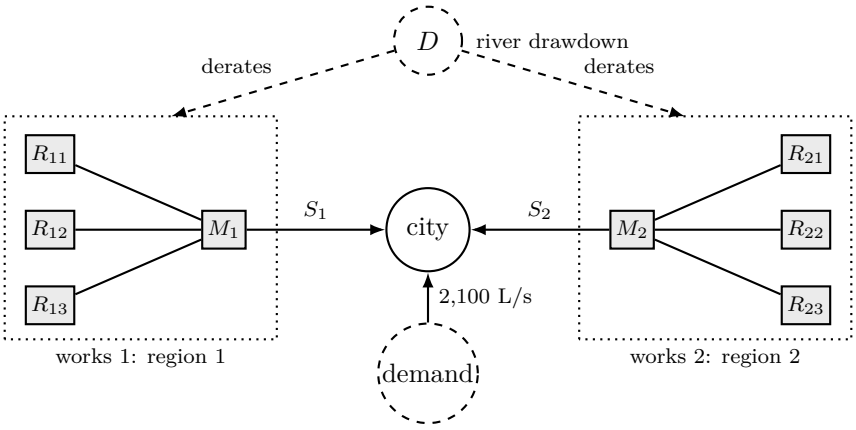


Figure 10.3: Two treatment works serving one city. Each works has three treatment streams in parallel, outputs R_{kj}, and a transfer main whose capacity M_k caps what it delivers; S_k is the works' delivered flow and its port to the city. The river drawdown D derates every stream at both works. Given D the works are independent, so the separator between them is $\{D, S_1, S_2\}$, not the ports alone.

and the city's balance leaves the rest unserved:

$$S_k = \min\Big((1-\alpha D)\sum_{j=1}^{3} R_{kj},\ M_k\Big), \qquad U = \big(2{,}100 - S_1 - S_2\big)^{+}. \tag{10.4}$$

The unserved demand U is water the city asks for and does not get, and it shows up as falling pressure and, past a point, as a hosepipe ban. There are nine uncertain inputs: six stream outputs, two main capacities and the drawdown. Figure 10.3 draws the works and the cut.

The shared river is the whole point of the example. A utility that models its two works separately, each with its own uncertainty, is making exactly the mistake Proposition 10.5 names below: it gets the mean right and the risk badly wrong, because the drought that derates one works derates the other in the same breath.

The dependency graph has the shape of the three-main model's, twice over, with one node more. Inside each works the stream outputs enter only through their sum, a cut of one dimension. The works' delivered flow is its port to the city, and the drawdown feeds both works. Given the drawdown the works are independent, and each works' message to the city is the distribution of its delivered flow.

Proposition 10.5 (Region messages with a shared input)**.** *Let the inputs be (W, X_1, X_2), mutually independent, let region k's port be $S_k = s_k(W, X_k)$, and let $G = h(S_1, S_2)$. Then*

$$\mathbb{E}[\phi(G)] \;=\; \sum_{w} p(w) \sum_{s_1, s_2} \phi\big(h(s_1,s_2)\big)\, q_1(s_1 \mid w)\, q_2(s_2 \mid w), \qquad q_k(s \mid w) = \sum_{x_k:\ s_k(w,x_k)=s} p_k(x_k). \tag{10.5}$$

The cost is one region solve per distinct value of each region's port for each value of W, and the combination is an evaluation of h. If the two regions are identical in law, $\mathrm{Cov}(S_1, S_2) = \mathrm{Var}(\mathbb{E}[S_1 \mid W])$*. The product of the marginal messages, $q_1(s_1)\, q_2(s_2)$, which is what the computation gives when W is left out of the separator, implies a covariance of zero, and is wrong whenever $\mathbb{E}[S_1 \mid W]$ varies with W.*

Table 10.4: The two treatment works. For each grid size: the solves of the full product grid on nine inputs; the works solves of the region route of Proposition 10.5; the errors of $\mathbb{E}[U]$ and $\mathbb{P}(U > 0)$ against the reference; and the rms error of Monte Carlo on $\mathbb{E}[U]$ at as many full solves as the region route's works solves, by Proposition 10.2.

n	full grid	works solves	$\mathbb{E}[U]$ error [L/s]	$\mathbb{P}(U > 0)$ error	Monte Carlo rms [L/s]
5	2.0×10^6	650	2.34	0.0118	1.08
9	3.9×10^8	4,050	0.593	0.0068	0.433
17	1.2×10^{11}	28,322	0.146	0.0011	0.164
33	4.6×10^{13}	211,266	0.035	0.0007	0.060

Proof. Condition on $W = w$. The inputs X_1 and X_2 are independent of each other and of W, so given $W = w$ the ports $s_1(w, X_1)$ and $s_2(w, X_2)$ are functions of independent variables and are independent. Their distributions are $q_1(\cdot \mid w)$ and $q_2(\cdot \mid w)$ by Proposition 10.4 applied inside each region, which gives $\mathbb{E}[\phi(G) \mid W = w]$ as the inner double sum; averaging over w gives (10.5). For the covariance, the law of total covariance gives $\mathrm{Cov}(S_1, S_2) = \mathbb{E}[\mathrm{Cov}(S_1, S_2 \mid W)] + \mathrm{Cov}(\mathbb{E}[S_1 \mid W], \mathbb{E}[S_2 \mid W])$. The first term is zero by conditional independence. When the regions are identical in law, $\mathbb{E}[S_1 \mid W] = \mathbb{E}[S_2 \mid W]$, and the second term is the variance of that function of W. □

Table 10.4 compares the routes. The full product grid on nine inputs is n^9 solves, 3.9×10^8 at $n = 9$ and 1.2×10^{11} at $n = 17$, and nobody runs it. Proposition 10.5 replaces it by $2n^2(3n - 2)$ works solves, 4,050 at $n = 9$. For each of the n drawdown values each works is enumerated on its own cut: $3n - 2$ values of the output sum, from two convolutions, times n main capacities. The combination $U = (2{,}100 - S_1 - S_2)^+$ is the city's balance, evaluated by sorting one message against the other, with no physics in it. At $n = 5$ the full grid is still affordable, 1,953,125 solves, and it gives $\mathbb{E}[U] = 12.588677026$ litres a second; the region route gives the same number from 650 works solves, to 2.5×10^{-14}. The reference is a region computation at $n = 161$: $\mathbb{E}[U] = 10.248$ litres a second, $\mathbb{P}(U > 0) = 0.195$ and a standard deviation of 27.6. Twenty million Monte Carlo draws agree with it to 0.008 litres a second.

Read the last two columns of the table together, and the region route does not beat sampling on the mean. At $n = 9$ its error of 0.59 litres a second is what Monte Carlo reaches in about 2,200 draws. At $n = 17$ its 0.146 would take Monte Carlo about 36,000 draws, against 28,322 works solves, and a works solve is about half a full solve. The probability of a shortfall tells the same story: at 4,050 draws its Monte Carlo standard error is 0.0062, against the region route's 0.0068. The reason is Proposition 10.1 in five effective dimensions, the drawdown and two per works once the stream outputs are summed. That is past the crossover at four, and the unserved demand is a tail quantity, zero over four fifths of the input space and bending sharply at its edge. The error is spread over the directions: nine drawdown nodes with 33 points on the other inputs still leave 0.40 litres a second, and 33 drawdown nodes with nine points on the others leave 0.23. The cut made the grid possible at all, 4,050 solves in place of 3.9×10^8. It did not make the grid better than sampling for one number. That is the three-main lesson read the other way: quadrature's advantage in three dimensions was not the book's, and its absence in five is not the book's failure.

Now ask the question a duty manager asks: how bad is it in a drought? The region computation already holds the answer. Its outer sum runs over the drawdown, and before that

sum is taken each term is $\mathbb{E}[U \mid D = d]$, Proposition 10.3 with $u = \{D\}$. Figure 10.4 draws it. Below 3 centimetres of drawdown the works never fall short. At 6.25 centimetres the expected shortfall is 4.5 litres a second with a 15 percent chance of any; at 10 centimetres it is 66 litres a second with an 84 percent chance. The nine values cost nothing beyond the 4,050 works solves, and they are within 0.27 litres a second rms of the fine computation. A Monte Carlo estimate of the same curve sorts its draws into nine drawdown bins and averages each bin. At the same 4,050 draws each bin holds about 450, and the rms error of the nine bin means over eight seeds is 0.97 litres a second, three and a half times the region route's. Since the error falls as the inverse square root of the draws, matching the region route would take about thirteen times the solves. The river alone accounts for 39 percent of the variance of U, a first-order attribution in the sense of Chapter 14, and it is again a partial sum.

Tails are a harder test. The probability that the shortfall exceeds t is $\mathbb{P}(S_1 + S_2 < 2{,}100 - t)$: the same messages with the balance evaluated at a lower load, so it too costs no further solve. Its accuracy is another matter. At $t = 100$ litres a second the reference probability is 0.0269. The region route gives 0.0325 at $n = 9$ and 0.0281 at $n = 17$, errors of 21 and 4 percent, while Monte Carlo at 4,050 draws has a relative standard error of 9.5 percent. At $t = 150$ litres a second, a probability of 0.005, the region route is off by 25 and 16 percent and Monte Carlo by 22 percent at 4,050 draws and 8 percent at 28,322. On a tail probability the grid does no better than sampling, and the reason is Proposition 10.1 again, this time by its failure. A probability is the expectation of an indicator, which jumps rather than bends. On a cell containing a jump the trapezoid rule is off by up to half the jump times h, so the error is of order h, not h^2. The remedy is to integrate one input in closed form inside the last region, which turns the jump back into a kink (Exercise 10.7).

Last, the separator. Leave the drawdown out of it: compute each works' message averaged over the river, and combine the two as independent regions. Every number comes out too small. At $n = 33$ the expected shortfall is 5.8 litres a second against 10.3, the probability of one is 0.145 against 0.195, and the standard deviation is 19.1 against 27.6. Proposition 10.5 says why. One works' delivered flow has mean 1,092 litres a second and standard deviation 56.0; the part of its variance that the drawdown explains is 1,324 in the same squared units, so the two works' deliveries are correlated at 0.42. The standard deviation of their sum is 94.5 litres a second with the shared drawdown and 79.2 without it. The unserved demand is the tail of that sum below 2,100, so a thinner tail means a smaller shortfall. A dry spell derates both works at once. A model that lets one works' bad week be offset by the other's good one has forgotten that there is only one river.

Principle 10.2. *A common cause belongs in the separator. An input that feeds two regions, such as the weather, a shared supply or a common-mode failure, must be conditioned on before the regions' messages are combined. Combining their marginal messages treats the regions as independent and understates every joint risk.*

10.8 Beyond three inputs

The product grid dies at n^d. With fourteen uncertain inputs and nine points each it is 2×10^{13} solves, and there is no cut of dimension two to save it. What survives is the structure, and it survives in two forms.

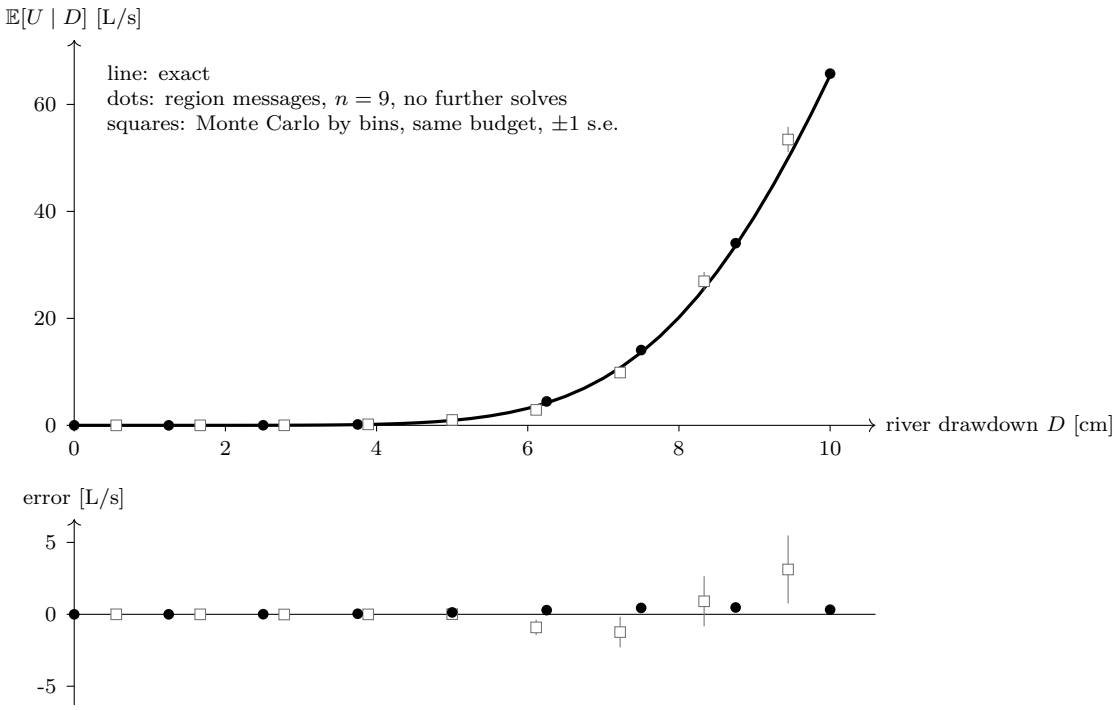


Figure 10.4: The expected unserved demand given the river drawdown, for the two treatment works. Top: the curve is exact, from a region computation at $n = 33$. The dots are the nine values held by the region computation at $n = 9$, which cost nothing beyond its 4,050 works solves. The squares are Monte Carlo at the same 4,050 draws sorted into nine drawdown bins, with one standard error, for one seed of the eight behind the rms error quoted in the text. Bottom: the errors of both against the exact curve, on a scale ten times finer.

The first is Chapter 9's simulation inside a region: sample the inputs of each region, solve each region on its own variables, and combine the regions' particle messages exactly on their ports. The samples are per region, the combination is on the separator, and the solves are of regions rather than of the whole. Chapter 18 measures what that saves on a real network.

The second is to make each solve return more than a number. When the physics is solved once and returns, along with the shortfall, its gradient with respect to every uncertain input, the solves form a census of value-and-gradient pairs from which many questions can be answered without further solves. Chapter 14 develops this, and a companion census study in the power domain, run with the same engine and committed to the repository that accompanies the book, runs it on a network with 73 buses and 14 uncertain line ratings, every rating set at 50 percent of its nominal value so that the network is stressed, and its committed results give the count. A census of 3,000 solves, certified on 2,000 further holdout draws, plus 512 for the mean and 512 for each of two questions that needed a small residual sample, answered the mean and nine planning questions: the marginal value of each line's capacity, the attribution of variance to each rating, the value of a measurement, a screening of which capacities matter, the mean under a changed weather belief, the mean under correlated capacities, a what-if, the tail, and the worst case. The total was 6,536 solves. The same nine questions answered by the standard sampling method for each cost 113,876; each has its own sample, except the value of a measurement, which reads the attribution's sample at no further solves. Seven of the nine cost the census route nothing at all

beyond the census. That is Principle 10.1 at scale: one computation held in a form that the next question can read.

10.9 The claim

There are three claims one could make for the factorized route, in increasing strength, and the book makes the third.

Weak. The factorized route handles combinatorial uncertainty without enumerating every state. True, and worthless: sampling does too.

Strong. The factorized route exploits the physical factorization and its separators to reuse expensive physics solves across many uncertain states and many queries. True, and established in this chapter: the cut ladder reuses solves across states, the product set reuses them across queries, and the census reuses them across nine questions.

Strongest. For equivalent accuracy, the number of expensive physics solves the factorized route needs is much smaller than the number sampling needs, and the ratio grows with the number of questions asked. Measured: about twenty-eight to one for one question in three dimensions, which is quadrature's doing and which disappears on the nine inputs of the two treatment works; no further solves against $6N$ for the second question; about thirteen to one for the works' shortfall given the river; seventeen to one over the mean and nine questions on a real network. The last three are reuse's doing.

The strongest claim carries its conditions. The reuse across states needs a cut of small dimension, which a physical system usually has and a generic function of many inputs usually does not. The reuse across queries needs the computation to be held in a form the next query can read, a product set, a region message, a census with gradients, rather than a single number. And the comparison is in solves; where solves are cheap, sampling's simplicity may be worth more than its solve count. Where they are not, which is every network of operational size, the count is what matters, and it is the count the book keeps.

10.10 In practice

Count solves, at a stated accuracy. Wall-clock time depends on the machine and the implementation, and state counts depend on the discretization. The number of physics solves needed to reach a stated error on a stated question is the comparison that transfers. Plot error against solves, as Figure 10.2 does, not error against grid size or sample size.

Read the cuts from the compiled graph. The dependency graph of the compiled model records what each quantity reads. Walk it upward from the quantity of interest and note the frontier at each level. The smallest frontier is the cheapest exact enumeration, and a frontier component that is a sum of independent inputs is a convolution, which costs no solve.

Put every shared input in the separator. Before combining region messages, list the inputs that feed more than one region: weather, a shared source or supply head, a common maintenance crew, a common-mode failure. Condition on each. The test is mechanical: an input with edges into two regions is a separator variable, whatever its physical scale.

Check exactness once, on a grid small enough to enumerate. The two-works example checked the region route against 1.95 million full solves at $n = 5$ before trusting it at $n = 33$. The check is cheap at small n and catches a misassigned input, which is the usual error.

Match the route to the question. For one mean in many effective dimensions, sampling is simple and competitive, and it should be used. For conditional means, attributions and many questions of one model, hold the computation in a form the next question can read: the product set, the region messages, the census.

Refine where the error is. A grid's error has a part per coordinate, and refining one coordinate costs in proportion to its nodes, not to the whole grid. In the region route the outer coordinate is the cheapest to refine: refining only the drawdown, from nine nodes to 33, cuts the works' error of the mean from 0.59 to 0.23 litres a second at 14,850 works solves.

10.11 What is missing

Every query in this chapter was a forward one: uncertain inputs pushed through the physics to a consequence. Chapter 11 turns to the inverse direction under operation, where readings arrive one at a time and the question is how cheaply each can be folded into the posterior; the answer, a rank-one update, is the cheapest reuse in the book. Part IV then asks the questions an operator asks, and Chapter 14 in particular develops the census with gradients that §10.8 previewed.

Notes and further reading

The three-main comparison and the cut ladder are a companion case study in the power domain, committed to the repository that accompanies the book, and this chapter's numbers are its committed results read verbatim. The hydraulic counterpart of its cost claim is Chapter 21, which ranks ninety-two leak hypotheses on a water network by two routes, one costing a single solve and the other ninety-three, and reports what the ninety-two extra solves buy. The advantage of quadrature over sampling in low dimension, and the curse of dimensionality that ends it, are standard; Robert and Casella [76] treat both sides. Variance-based attribution and the Shapley share are Sobol's [85], Owen's [67] and Song, Nelson and Staum's [86]; the pick-freeze estimator and its $(d + 4)N$ or $7N$ solve count for $d = 3$ are Saltelli's [81]. That every conditional expectation is a partial sum on a product grid is Fubini's theorem and needs no citation; that the model's compiled dependency graph exposes the cuts on which the sum can be taken is the framework's. The two treatment works of §10.7 were built for this chapter; their constants are typical of a mid-size supply zone rather than taken from one. The law of total covariance behind Proposition 10.5 and the trapezoid error bound behind Proposition 10.1 are standard, and both are proved on the page.

The three-strength statement of the claim in §10.9 is the book's, and it was arrived at by conceding the weak claim to sampling first. The reader who has met the literature on "probabilistic power flow" will recognize the weak claim as the one usually made there, and the strong one as the one that point-estimate, cumulant and surrogate methods make implicitly; the census with gradients of §10.8 is a surrogate method, and Chapter 14 says which.

Summary

- Sampling answers every question and does not care about the size of the state space; what it pays for is the solve. The comparison is in solves per question at equal accuracy.
- On three uncertain capacities, the product grid reaches at 729 solves an accuracy that sampling needs about 20,700 draws for: quadrature's advantage in low dimension, not the book's.
- The trapezoid grid's error is of order h^2 even across kinks, and Monte Carlo's is $\sigma/\sqrt{N}$ whatever the dimension. In solves the grid's exponent is $2/d$ against sampling's $1/2$.
- The attribution costs the grid nothing beyond the grid, because conditional expectations are partial sums on a product set; sampling needs $7N$ solves and is an order of magnitude less accurate at 21,000.
- The model's dependency graph has cuts; enumerating on the cut $(A{+}C, B)$ needs 561 solves for the same mean to 10^{-13} as the 4,913-solve grid. The exponent is the cut's dimension.
- Beyond three inputs the grid dies but the structure survives, as region messages and as a census with gradients: 6,536 solves for the mean and nine questions against 113,876 on a 73-bus network.
- On two treatment works with nine inputs, region messages conditioned on the shared drawdown replace n^9 solves by $2n^2(3n-2)$, exactly. They match sampling on the mean and beat it by about thirteen in solves on the shortfall given the river.
- A common cause belongs in the separator: combining the works' marginal messages understates the expected shortfall by more than forty percent.
- The book's claim is the strongest one, with its conditions: a small cut, a reusable form, and expensive solves.

Exercises

Exercise 10.1. Derive the closed form of $\mathbb{E}[G]$ for the three-main problem from equation (10.1) and the uniform priors, and confirm 16/3. Then derive $\mathbb{P}(G = 0)$ and confirm 0.46.

Exercise 10.2. Show that on an n-point uniform grid for A and for C, the sum $A + C$ takes $2n - 1$ distinct values, and derive their probabilities. Confirm the $(2n-1)\,n$ entries of the third row of Table 10.3.

Exercise 10.3. Implement pick-freeze for the three-main problem and reproduce the $7N$ solve count. Then implement the sampler that draws $A + C$ from its convolution and show how many of the seven sample sets it saves.

Exercise 10.4. The grid's mean error falls as n^{-2} and Monte Carlo's as $N^{-1/2}$. For d uncertain inputs the grid costs n^d solves. Find the dimension at which, for an error of 0.01 litres a second on a problem with the same constants as this one, sampling first needs fewer solves than the product grid.

Exercise 10.5. For the two-district network of Chapter 8 with all five demands uncertain, list the cuts of its dependency graph in nodal form, their dimensions, and the distinct solves each needs on an n-point grid. Which cut is the port pair, and what does enumerating on it cost?

Exercise 10.6. For the two treatment works of §10.7, derive $\mathbb{E}[S_k \mid D]$ in closed form when the transfer main never binds, and from it $\mathrm{Var}(\mathbb{E}[S_k \mid D])$. Compare with the 1,324 computed with the main in place, and explain the sign of the difference.

Exercise 10.7. In the region computation for $\mathbb{P}(U > t)$, replace works 2's grid over its main capacity by the exact uniform distribution, so that given the drawdown, works 2's output sum and works 1's delivery, the probability of a shortfall above t is a continuous, piecewise-linear function of the remaining inputs. Show that the error in $\mathbb{P}(U > 100)$ then falls as h^2, and find the n at which it beats Monte Carlo at the same number of works solves.

Solutions to selected exercises

Exercise 10.1. Write $S = A + C$. Because S and B are both at least 140, $\min(S, B) \geq 140$ and G never exceeds 20. For $0 \leq t < 20$, $G > t$ exactly when $\min(S, B) < 160 - t$, which is the complement of the event that both $S \geq 160 - t$ and $B \geq 160 - t$. The capacities are independent, so

$$\mathbb{P}(G > t) = 1 - \mathbb{P}(S \geq 160 - t)\,\mathbb{P}(B \geq 160 - t).$$

B is uniform on $[140, 180]$, so $\mathbb{P}(B \geq 160 - t) = (20 + t)/40$. The sum of two independent uniforms on $[70, 120]$ has the triangular density $(s - 140)/2500$ on $[140, 190]$, so $\mathbb{P}(S < 160 - t) = (20 - t)^2/5000$. The mean of a nonnegative variable is the integral of its tail:

$$\mathbb{E}[G] = \int_0^{20} \Big[1 - \frac{20 + t}{40} + \frac{(20 - t)^2(20 + t)}{200{,}000}\Big]\,dt = 20 - 15 + \tfrac{1}{3} = \tfrac{16}{3},$$

the last integral by $u = 20 - t$: $\int_0^{20} u^2(40 - u)\,du/200{,}000 = (320{,}000/3 - 40{,}000)/200{,}000 = 1/3$. At $t = 0$ the tail is $1 - 0.92 \times 0.5 = 0.54$, so $\mathbb{P}(G = 0) = 0.46$. The same route gives the second moment, $\mathbb{E}[G^2] = \int_0^{20} 2t\,\mathbb{P}(G > t)\,dt = 400 - 1000/3 + 56/15 = 70.4$, and the variance $70.4 - (16/3)^2 = 41.956$ quoted in §10.2.

Exercise 10.2. On the n-point grid the capacities of A and C take the values $70 + ih$, $i = 0, \ldots, n - 1$, with $h = 50/(n - 1)$. A sum $a_i + c_j = 140 + (i + j)h$ depends only on $i + j$, which runs over $0, \ldots, 2n - 2$: $2n - 1$ distinct values. By Proposition 10.4 the probability of the k-th is $\sum_{i+j=k} w_i w_j$, the discrete convolution of the weight vector with itself. With the trapezoid weights, $w_0 = w_{n-1} = 1/(2(n - 1))$ and $w_i = 1/(n - 1)$ otherwise, the result is a discrete triangle with ramps at its ends, $(4, 16, 32, 48, 56, 48, 32, 16, 4)/256$ at $n = 5$. The capacity of B takes n values independently of $A + C$, so the pair $(A + C, B)$ takes $(2n - 1)n$ values: 153, 561, 1,225, 2,145 and 3,321 at $n = 9$, 17, 25, 33 and 41, the third row of Table 10.3. Computing $\mathbb{E}[G]$ on those pairs, with the convolved weights times B's weights, reproduces the full-grid means of Table 10.1 to 2×10^{-15}.

Chapter 11

Sequential evidence

Every posterior so far was computed with all the readings in hand. In operation they are not. A sensor reports, then another, then the first again a minute later, and the question after each is what the system now believes. Re-solving the whole model after every reading is correct and wasteful: the reading is one row, and one row changes the posterior by a rank-one term. This chapter shows that folding a reading into a Gaussian posterior costs one solve against a factorization already held, that the same solve yields the number by which the reading surprised the model, that the value of a reading can be computed before it is taken, and that at the scale of a real model the update is tens of times cheaper than a re-solve. It closes Part III.

11.1 Readings arrive one at a time

Take the posterior of a linear-Gaussian branch after the readings so far: precision $\mathbf{\Lambda}$, information $\boldsymbol{\eta}$, mean $\boldsymbol{\mu} = \mathbf{\Lambda}^{-1}\boldsymbol{\eta}$, covariance $\mathbf{\Sigma} = \mathbf{\Lambda}^{-1}$, with $\mathbf{\Lambda}$ held as a sparse factorization. A new reading y of the linear combination $\mathbf{h}^{\mathsf{T}}\mathbf{z}$ with accuracy σ_y is one more row of $\mathbf{g}$, and by §6.1 it adds $w\,\mathbf{h}\mathbf{h}^{\mathsf{T}}$ to the precision and $w\,y\,\mathbf{h}$ to the information, $w = 1/\sigma_y^2$. For a sensor on one variable, $\mathbf{h}$ is a unit vector and the change is one diagonal entry. The new posterior could be had by refactoring the new precision. It need not be.

11.2 One reading, one rank-one update

Proposition 11.1 (Rank-one conditioning)**.** *Let* $\mathbf{z} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{\Sigma})$ *and let* $y = \mathbf{h}^{\mathsf{T}}\mathbf{z} + \nu$ *with* $\nu \sim \mathcal{N}(0, \sigma_y^2)$ *independent of* $\mathbf{z}$*. Then* $\mathbf{z} \mid y$ *is Gaussian with*

$$\boldsymbol{\mu}' = \boldsymbol{\mu} + \mathbf{k}\,(y - \mathbf{h}^{\mathsf{T}}\boldsymbol{\mu}), \qquad \mathbf{\Sigma}' = \mathbf{\Sigma} - \mathbf{k}\,\mathbf{s}^{\mathsf{T}}, \qquad \mathbf{s} = \mathbf{\Sigma}\mathbf{h}, \quad \mathbf{k} = \frac{\mathbf{s}}{\sigma_y^2 + \mathbf{h}^{\mathsf{T}}\mathbf{s}}. \tag{11.1}$$

Proof. In information form the reading adds $w\mathbf{h}\mathbf{h}^{\mathsf{T}}$ to the precision and $wy\mathbf{h}$ to the information, $w = 1/\sigma_y^2$. Write $S = \sigma_y^2 + \mathbf{h}^{\mathsf{T}}\mathbf{s}$. The claim for the covariance is that $\mathbf{\Sigma}' = \mathbf{\Sigma} - \mathbf{s}\mathbf{s}^{\mathsf{T}}/S$ inverts $\mathbf{\Lambda}' = \mathbf{\Lambda} + w\mathbf{h}\mathbf{h}^{\mathsf{T}}$. Multiply, using $\mathbf{\Lambda}\mathbf{\Sigma} = \mathbf{I}$ and $\mathbf{\Lambda}\mathbf{s} = \mathbf{h}$:

$$\mathbf{\Lambda}'\mathbf{\Sigma}' \;=\; \mathbf{I} + w\,\mathbf{h}\mathbf{s}^{\mathsf{T}} - \frac{(1 + w\,\mathbf{h}^{\mathsf{T}}\mathbf{s})}{S}\,\mathbf{h}\mathbf{s}^{\mathsf{T}} \;=\; \mathbf{I} + w\,\mathbf{h}\mathbf{s}^{\mathsf{T}} - w\,\mathbf{h}\mathbf{s}^{\mathsf{T}} \;=\; \mathbf{I},$$

since $(1 + w\,\mathbf{h}^{\mathsf{T}}\mathbf{s})/S = (\sigma_y^2 + \mathbf{h}^{\mathsf{T}}\mathbf{s})/(\sigma_y^2 S) = w$. This is the Sherman–Morrison identity. For the mean, $\boldsymbol{\mu}' = \mathbf{\Sigma}'(\boldsymbol{\eta} + wy\mathbf{h})$. The first part is $\mathbf{\Sigma}'\boldsymbol{\eta} = \boldsymbol{\mu} - \mathbf{s}\,(\mathbf{h}^{\mathsf{T}}\boldsymbol{\mu})/S$. The second uses

$\boldsymbol{\Sigma}'\mathbf{h} = \mathbf{s} - \mathbf{s}\,(\mathbf{h}^\mathsf{T}\mathbf{s})/S = \mathbf{s}\,\sigma_y^2/S$, so $wy\,\boldsymbol{\Sigma}'\mathbf{h} = y\,\mathbf{s}/S$. Adding, $\boldsymbol{\mu}' = \boldsymbol{\mu} + \mathbf{s}\,(y - \mathbf{h}^\mathsf{T}\boldsymbol{\mu})/S$, which is $\boldsymbol{\mu} + \mathbf{k}\,(y - \mathbf{h}^\mathsf{T}\boldsymbol{\mu})$. □

Three things about equation (11.1) carry the chapter.

The only expensive object is $\mathbf{s} = \boldsymbol{\Sigma}\mathbf{h}$, and it is not computed by forming $\boldsymbol{\Sigma}$. It is the solution of $\boldsymbol{\Lambda}\mathbf{s} = \mathbf{h}$ against the factorization already held: two triangular solves, the same operation as one selected inversion in §6.3. For a sensor on a single variable, $\mathbf{s}$ is one column of the posterior covariance, the covariance of that variable with every other, which is what a reading of that variable needs to know in order to move every other.

The vector $\mathbf{k}$ is the *gain*. It says how far each variable moves per unit of surprise, and it is the covariance column scaled by the total uncertainty of the reading, prior plus sensor. A variable tightly correlated with the measured one moves a lot; an independent one does not move.

And the covariance update is a subtraction of a rank-one matrix that is never formed. Keep $\boldsymbol{\Sigma}$ implicit: hold the factorization of the original $\boldsymbol{\Lambda}$ and a list of the pairs $(\mathbf{k}_i, \mathbf{s}_i)$ from the updates so far. Then any product $\boldsymbol{\Sigma}'\mathbf{v}$ is a solve against the factorization followed by a subtraction per pair, and any marginal variance is a solve followed by a few inner products. After r readings each solve costs r extra inner products, which is nothing until r approaches the ratio of a factorization's cost to a solve's, at which point one refactors and empties the list.

Principle 11.1. *A reading is a rank-one change to the posterior. Folding it in costs one solve against the factorization already held, and the same solve yields the covariance column the reading needed and the gain it produced. Nothing is refactored.*

11.3 The innovation

The scalar $y - \mathbf{h}^\mathsf{T}\boldsymbol{\mu}$ in equation (11.1) is the *innovation*: the reading minus what the model predicted for it. Its predicted spread, $\sqrt{\sigma_y^2 + \mathbf{h}^\mathsf{T}\mathbf{s}}$, is the sensor's accuracy combined with the posterior's own uncertainty about the measured quantity, and the ratio of the two, the *standardized innovation*, is a number the model should see distributed as a standard Gaussian if the model is adequate. A reading that lands three or four spreads from its prediction is either a faulty sensor or a faulty model, and it is flagged before it is folded in. Chapter 12 builds detection on this; here it is enough that the number costs nothing beyond the update, because $\mathbf{h}^\mathsf{T}\mathbf{s}$ was computed for the gain.

Proposition 11.2 (Innovations)**.** *Let readings $y_1, \dots, y_r$ of an adequate linear-Gaussian model be folded in one at a time, and let e_i and s_i be the innovation of reading i and its predicted variance, both computed before reading i is folded in. Then the standardized innovations $e_i/\sqrt{s_i}$ are independent standard Gaussians, $\sum_i e_i^2/s_i$ is chi-squared with r degrees of freedom, and the log predictive density of all the readings is*

$$\log p(y_1, \dots, y_r) \;=\; -\tfrac{1}{2}\sum_{i=1}^{r}\Big(\log 2\pi s_i + \frac{e_i^2}{s_i}\Big). \tag{11.2}$$

Proof. By the chain rule of probability, $p(y_1, \dots, y_r) = \prod_i p(y_i \mid y_1, \dots, y_{i-1})$. Under the model, y_i given the earlier readings is Gaussian with mean $\mathbf{h}_i^\mathsf{T}\boldsymbol{\mu}_{i-1}$ and variance s_i: the prediction that Proposition 11.1 makes before it updates. So each factor is the Gaussian density of e_i with

variance s_i, and taking logs gives (11.2). For independence, $e_i/\sqrt{s_i}$ given $y_1, \ldots, y_{i-1}$ is standard Gaussian whatever those readings are. The earlier standardized innovations are functions of them, so $e_i/\sqrt{s_i}$ is standard Gaussian given the earlier innovations too. A sequence each member of which is standard Gaussian given all the earlier ones has a joint density that factors into standard Gaussians, which is independence. The sum of squares of r independent standard Gaussians is chi-squared with r degrees of freedom by definition. □

Equation (11.2) is the prediction-error decomposition. The predictive density that Proposition 7.1 used to weigh branches accumulates one reading at a time, from the two numbers each update computes anyway.

The chain, one reading at a time. Return to the flow chain of Figure 3.1 with hard physics. Every number is from the script `ch11_sequential.py` in the book's `scripts` directory. Under the prior, h_2 is predicted at 70.00 ± 10.45. The reading is 72.0: innovation +2.00, standardized +0.19, unremarkable. One rank-one update gives $G = 103.66 \pm 5.82$ and $h_a = 20.16 \pm 2.87$, the second column of Table 3.2. Now h_1 is predicted at 92.73 ± 1.43, the virtual sensor of §3.4 with its spread now including the instrument's. The reading is 93.0: innovation +0.27, standardized +0.19 again. The second update gives $G = 104.63 \pm 2.86$, $h_a = 19.73 \pm 1.73$, the third column. The batch posterior with both readings agrees with the two sequential updates to eight decimals in the mean and seven in the covariance, the residual being the conditioning of the tight physics rows and not the method.

Figure 11.1 adds a third reading, $h_a = 19.5$ with accuracy 0.5, and folds the three in three orders. In every order the source ends at 104.97 ± 0.97, the batch posterior with all three readings. The paths differ. Read the trunk main first and the source does not move at all, 100.00 ± 20.00, because under the prior the trunk main and the source are independent and the reading carries nothing about G. Read h_2 next and the source drops to ± 1.40, far better than the ± 5.82 that h_2 gave alone, because the trunk main reading has pinned the one quantity that h_2 confounds with the source. Read h_1 first and the source is at 104.09 ± 4.25, after which the room removes 94 percent of what remains. No innovation in any of the three orders exceeds a quarter of its spread, as it should not for readings consistent with the model.

11.4 Many readings

Readings in sequence are updates in sequence, and for a linear-Gaussian model the result does not depend on the order, since each update is exact and the final posterior is the one with all rows present. A block of r readings taken together is the Woodbury identity in place of Sherman–Morrison: r solves, an $r \times r$ dense system, the same result.

This is the Kalman filter with a static state. The filter of Chapter 16 adds a prediction step between updates, in which the state moves and its uncertainty grows; with no dynamics the prediction is the identity and only the update remains, and the update is equation (11.1). The reader who knows the filter will recognize $\mathbf{k}$ as the Kalman gain and $\sigma_y^2 + \mathbf{h}^\mathsf{T}\mathbf{s}$ as the innovation covariance. What the sparse factorization adds is that the state may have a hundred thousand components and the update still costs one solve, because $\boldsymbol{\Sigma}$ is never formed. The block form is derived in the solution to Exercise 11.1.

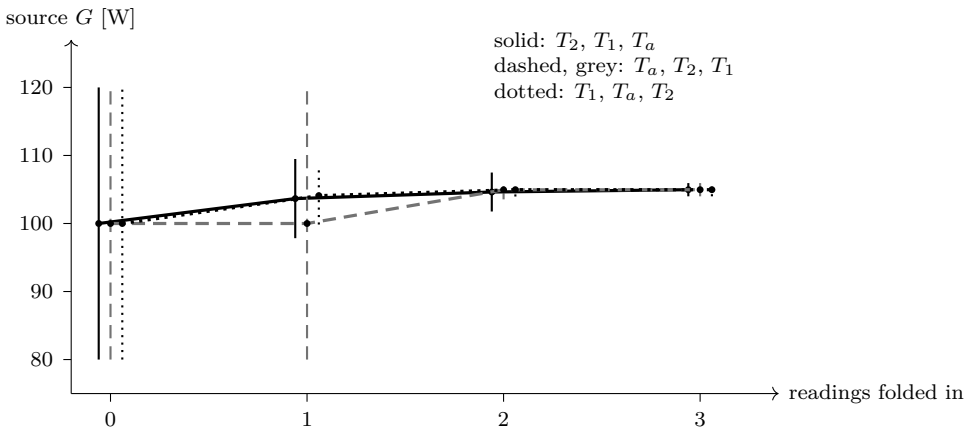


Figure 11.1: The source of the pumping chain as three readings are folded in, in three orders: $h_2 = 72.0$, $h_1 = 93.0$ and $h_a = 19.5$, each of accuracy 0.5. Dots are posterior means and bars one posterior standard deviation; the three orders are offset slightly for legibility. Every order ends at the batch posterior. Read first, the trunk main moves nothing, because under the prior the trunk main and the source are independent.

11.5 Worked example: a stream of flow readings

The two-district network of Chapter 8 carries six flow meters, on pipes R–2, 2–3, 3–4, 4–5, 4–6 and 5–6, each of accuracy 0.2 kg/s. They report in rounds, one after another, and the demands do not change between rounds: twenty-four readings of a static state. The state is a draw from the prior, with demands at junctions 2 through 6 of 13.62, 7.88, 1.33, 12.51 and 2.85 kg/s. From the third round the meter on pipe 4–5 reads 1.0 kg/s high, a calibration drift of five times its accuracy — which is what an electromagnetic meter does when its liner fouls. Every number is from the script `ch11_more.py`.

Figure 11.2 plots the standardized innovation of each reading and, below it, their running sum of squares against the 99 percent point of the chi-squared distribution with as many metres of freedom as readings. In the first round the innovations are large in the units of the readings, up to 1.7 kg/s, and ordinary in their own, none beyond 1.6 spreads, because the predicted spread of a first reading is mostly the prior's uncertainty about the flow: 1.50 kg/s for pipe R–2. By the second round every flow is known to about the meters' accuracy, and the predicted spreads have fallen to between 0.23 and 0.28. The drifting meter's first biased reading, the sixteenth, lands 4.6 spreads high, and its second, the twenty-second, 4.2. No other reading in the stream exceeds 1.6.

The running sum tells the same story less sharply. It sits at 15.5 against a quantile of 30.6 at the fifteenth reading and jumps to 36.9 against 32.0 at the sixteenth, the single bad reading carrying it over in one step; thereafter it stays above, because each clean reading adds less than the quantile grows. A cumulative test spreads a sudden fault over every reading that follows and would have taken until the sixteenth to say anything at all; the per-reading test sees it at the sixteenth too, but says *which* reading. The cumulative test earns its place on a slow drift that no single reading reveals, not on this one.

Detection matters because a folded-in fault corrupts everything the reading touches. Folding all twenty-four readings in puts the demand at junction 5 at 12.740 ± 0.137 kg/s, 1.7 standard deviations above the true 12.514. Withholding the two flagged readings gives 12.333 ± 0.151,

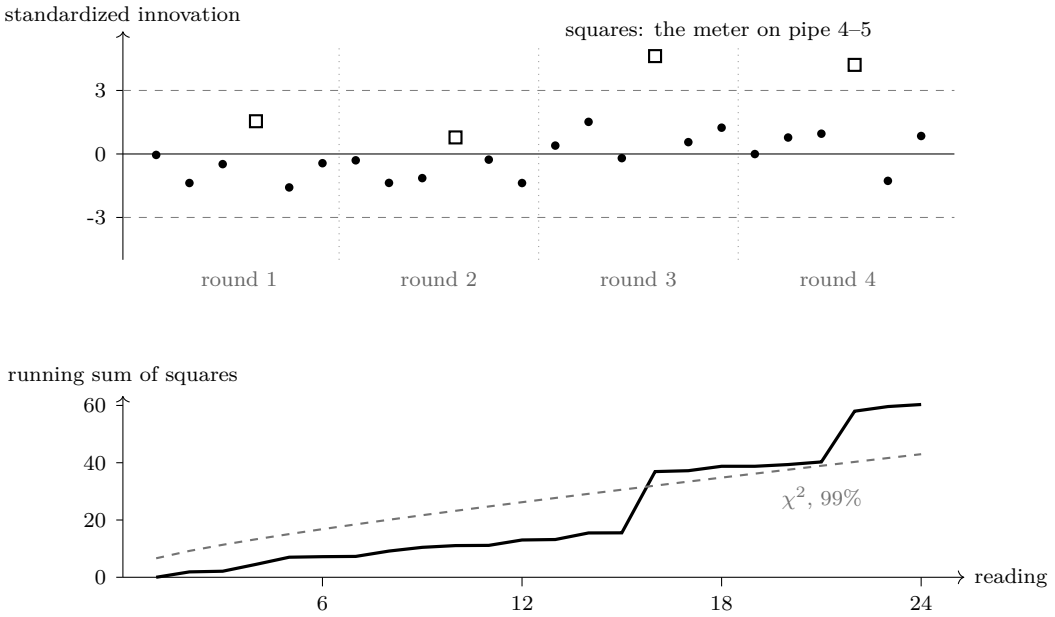


Figure 11.2: The two-district network's six flow meters read in four rounds, a static state. Top: the standardized innovation of each reading; squares mark the meter on pipe 4–5, which reads 1.0 kg/s high from the third round, and the dashed lines are at ±3. Bottom: the running sum of squared standardized innovations (solid) against the 99 percent point of the chi-squared distribution with as many degrees of freedom as readings (dashed).

1.2 below it. The rule that did the withholding, to hold aside any reading whose standardized innovation exceeds three, used nothing the update had not already computed. Chapter 12 turns the rule into a test with a stated false-alarm rate and asks which fault explains a flag.

Last, the evidence. The sum over the stream of the per-reading terms of equation (11.2) is −26.631625. The log predictive density of all twenty-four readings at once, a twenty-four-dimensional Gaussian with its own determinant and quadratic form, is the same to the six decimals shown.

11.6 The Laplace posterior and re-linearization

For a nonlinear model the posterior is Gaussian only in the Laplace sense of §6.4: the Gaussian at the mode with the curvature there. Proposition 11.1 conditions that Gaussian exactly. What it does not do is move the linearization point: the new mode of the true posterior differs from $\boldsymbol{\mu}'$ by the amount the nonlinear rows curve between the old mode and the new mean.

The right procedure is the cheap one first. Fold the reading in by the rank-one update; check the standardized innovation and the size of the shift $\boldsymbol{\mu}' - \boldsymbol{\mu}$ against the posterior width; if both are modest, the linearization is still good and nothing more is needed. If either is large, take one Gauss–Newton step from $\boldsymbol{\mu}'$, which lands on the exact new mode when the rows are nearly linear across the shift, and refactor there. For a model whose rows are linear, which the linearized water loop and the hydraulic models of this book are, the check always passes and the update is always exact. For a model carrying the real head-loss exponent the check fails when a reading moves a flow by a sizable fraction of the range over which $m^{1.852}$ curves — which, because that

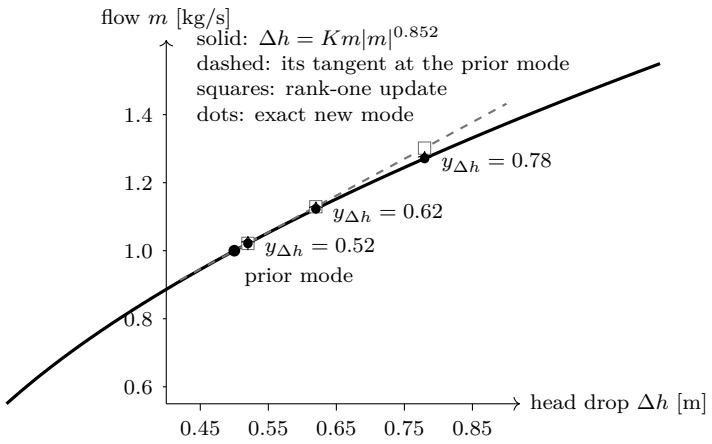


Figure 11.3: A pipe with head drop $\Delta h = K\,m\,|m|^{0.852}$ and three gauge readings of that drop. The Laplace posterior before the readings lies along the tangent at the prior mode (dashed). The rank-one update of that Gaussian moves along the tangent (squares), while the exact new modes lie on the curve (dots). The error grows with the shift, from a thirtieth to two and then eleven posterior standard deviations. One Gauss–Newton step from each square reaches its dot.

exponent bends hardest near zero flow, is exactly in the low-flow hours a leak search cares about.

A worked example on a nonlinear pipe. Take one pipe whose head drop is $\Delta h = K\,m\,|m|^{0.852}$ with K chosen so that it loses half a metre at the prior mean flow, a prior of 1.0 ± 0.3 kg/s on the flow it carries, a physics row of width 0.01, and a differential gauge across it reading the head drop to 0.01 m. Before any reading the Laplace posterior sits at $m = 1.000 \pm 0.300$ kg/s and $\Delta h = 0.500 \pm 0.278$ m, and the two are correlated at 0.9994: the Gaussian lies along the tangent to the head-loss curve at the mode. Figure 11.3 draws three readings of the head drop, 0.52, 0.62 and 0.78 m. For the first, the rank-one update gives $m = 1.0215$ against the exact new mode 1.0213 ± 0.0150, an error of a seventieth of a posterior standard deviation. For the second it gives 1.1293 against 1.1229 ± 0.0138, half a standard deviation off. For the third it gives 1.3016 against 1.2709 ± 0.0124, two and a half off. The update moves along the tangent, and the head-loss curve has bent away from it. In every case one Gauss–Newton step from the updated mean lands on the exact mode to 3×10^{-4} kg/s or better.

The two checks of the procedure coincide here, because the reading is of the head drop itself and much sharper than the prior: the standardized innovations are 0.07, 0.43 and 1.01, and so are the shifts in units of the prior's standard deviation of Δh. Not one of them would raise an alarm as a fault — a gauge reading one prior spread from its prediction is an ordinary Tuesday. The shift is what matters, and a shift of one prior standard deviation already cost two and a half posterior ones. Re-linearize whenever the shift exceeds about a quarter of the prior's standard deviation along the reading; the step costs one factorization.

11.7 The value of a reading before it is taken

Look at the covariance update in equation (11.1): it does not contain y. The reduction in uncertainty that a reading brings is known before the reading is taken. For a target quantity $g = \mathbf{c}^\mathsf{T}\mathbf{z}$, the posterior variance after a reading of $\mathbf{h}^\mathsf{T}\mathbf{z}$ with accuracy σ_y is

$$\mathrm{Var}'(g) \;=\; \mathrm{Var}(g) - \frac{(\mathbf{c}^\mathsf{T}\boldsymbol{\Sigma}\mathbf{h})^2}{\sigma_y^2 + \mathbf{h}^\mathsf{T}\boldsymbol{\Sigma}\mathbf{h}}, \tag{11.3}$$

whatever the reading turns out to be. It follows from the covariance update of equation (11.1) by multiplying on both sides by $\mathbf{c}$: $\mathbf{c}^\mathsf{T}\boldsymbol{\Sigma}'\mathbf{c} = \mathbf{c}^\mathsf{T}\boldsymbol{\Sigma}\mathbf{c} - (\mathbf{c}^\mathsf{T}\mathbf{k})(\mathbf{s}^\mathsf{T}\mathbf{c})$, and $\mathbf{c}^\mathsf{T}\mathbf{k} = \mathbf{c}^\mathsf{T}\boldsymbol{\Sigma}\mathbf{h}/(\sigma_y^2 + \mathbf{h}^\mathsf{T}\boldsymbol{\Sigma}\mathbf{h})$. The reduction is the squared covariance between the target and the measured quantity, scaled by the measured quantity's total uncertainty. A sensor is valuable to a target exactly insofar as the two are correlated under the current posterior, and its value changes as the posterior does.

When the target is the whole state rather than one quantity, the natural measure is the reduction in entropy, which for a Gaussian is half the log of the ratio of determinants of the covariances before and after. A single reading gives

$$\tfrac{1}{2}\log\frac{\det\boldsymbol{\Sigma}}{\det\boldsymbol{\Sigma}'} \;=\; \tfrac{1}{2}\log\frac{\det\boldsymbol{\Lambda}'}{\det\boldsymbol{\Lambda}} \;=\; \tfrac{1}{2}\log\Big(1 + \frac{\mathbf{h}^\mathsf{T}\boldsymbol{\Sigma}\mathbf{h}}{\sigma_y^2}\Big). \tag{11.4}$$

The last step is the matrix determinant lemma: $\det(\boldsymbol{\Lambda} + w\mathbf{h}\mathbf{h}^\mathsf{T}) = \det\boldsymbol{\Lambda}\,\det(\mathbf{I} + w\boldsymbol{\Sigma}\mathbf{h}\mathbf{h}^\mathsf{T})$, and $\mathbf{I} + w\,\mathbf{s}\mathbf{h}^\mathsf{T}$ has the eigenvalue $1 + w\,\mathbf{h}^\mathsf{T}\mathbf{s}$ on $\mathbf{s}$ and the eigenvalue 1 on every vector orthogonal to $\mathbf{h}$, so its determinant is $1 + w\,\mathbf{h}^\mathsf{T}\mathbf{s}$. A reading is worth most to the whole state where the state's own uncertainty about the measured quantity is largest compared with the sensor's accuracy.

Table 11.1 applies it. On the chain, a sensor on the trunk head is worth nothing to the source under the prior: the two are independent, as Chapter 5 said, so their covariance is zero and equation (11.3) subtracts nothing. A sensor on either head is worth most of the source's variance, and a direct flow meter, even a poor one, nearly all of it. Now read h_2 and ask again. The source's spread is 5.82, and the sensor that would reduce it most is the one on the trunk main, by 94 percent, because the reading has correlated the source with the trunk main at -0.985 and a reading of one is now a reading of the other. The sensor that was worthless has become the best. That is the ridge of §3.4 read as a decision: the next sensor should be placed where the posterior's remaining uncertainty is concentrated, and the posterior says where.

On the two-district network the target is the demand at junction 5, and the candidates are flow meters on each pipe. The meter on the pipe into junction 5 from junction 4 is worth 64 percent of the demand's variance; the meter on the link 5–6 next to it, 44 percent; the meter on the transfer main, two junctions away, a third; the meter on the main R–2 in the other district, a twelfth. The ordering follows the graph, and equation (11.3) computes it from one solve per candidate against the factorization already held.

Figure 11.4 shows how the values change once the best meter has been read. After the meter on pipe 4–5, the meter on the link 5–6, worth 44 percent of the prior variance, is worth 95 percent of what remains, and the meter on pipe 4–6 is worth 62 percent; the meters in district A fall to under one percent. The balance at junction 5 makes the demand the difference of the two flows at the junction. With one flow known, the other is the demand, and the meter that reads it directly becomes decisive. The whole-state measure of equation (11.4) orders the meters differently. The meter on the transfer main is worth 2.85 nats to the whole state, the most of

Table 11.1: The value of the next sensor, computed before any reading. Top: the chain, target G, prior spread 20. Bottom: the two-region network, target q_5, prior spread 0.200, flow meters of accuracy 0.02.

chain, sensor on	sd of G after	reduced by
under the prior		
h_1 (acc. 0.5)	4.25	95%
h_2 (acc. 0.5)	5.82	92%
h_a (acc. 0.5)	20.00	0%
m_{2a} (acc. 2.0)	1.99	99%
after h_2 is read (sd 5.82)		
h_1	2.86	76%
h_a	1.40	94%
m_{2a}	1.89	89%

network, meter on	sd of q_5 after	reduced by
m_{12}	0.187	12%
m_{23}	0.179	20%
m_{34}	0.164	33%
m_{45}	0.093	78%
m_{46}	0.179	20%
m_{56}	0.145	48%

the six, and the meter on pipe 4–5 only 2.02. The target chooses the sensor; there is no best sensor in the abstract.

Choosing sensors by this criterion, one at a time, each chosen given the last, is greedy sensor placement, and the criterion is the expected reduction in a target's variance. Chapter 12 generalizes the criterion to the expected information gain about the whole state, which is the classical measure of an experiment's value, and shows that for a Gaussian it is the log of a determinant ratio, computed by the same solves.

Principle 11.2. *The uncertainty a reading removes is known before the reading is taken. It is the squared covariance between target and measurement, divided by the measurement's total spread, and it is one solve per candidate.*

11.8 Cost at scale

The chain has six unknowns and the comparison is not worth timing. Take instead a grid network of $m \times m$ nodes, each conducting to its four neighbors and losing heat to an room reservoir, with an uncertain source at every node; eliminating the sources leaves one soft row per node on the heads, and the precision is sparse with the grid's own structure, which in two dimensions has fill. Ten head sensors at random nodes report in turn. Table 11.2 times the two ways of folding them in.

One rank-one update costs between a thirtieth and a seventieth of a factorization across three decades of size, the ratio being that of a solve to a factorization, which for a two-dimensional

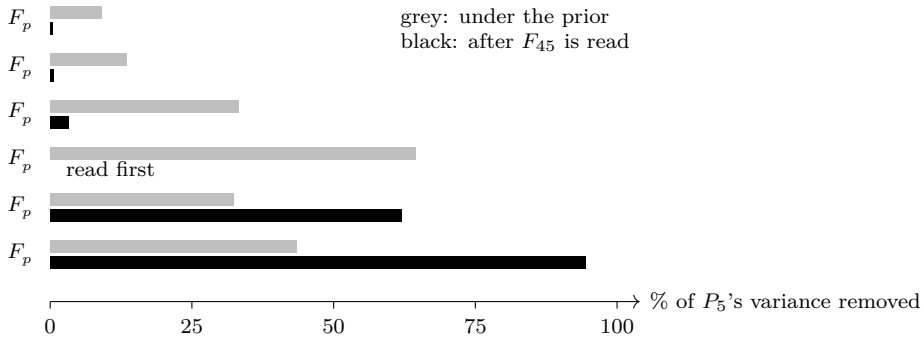


Figure 11.4: The value of each flow meter to the demand at junction 5, as the percentage of its current variance the meter would remove, computed before any reading. Grey: under the prior. Black: after the meter on pipe 4–5 has been read. The meter on the link 5–6 goes from second to decisive, because the demand at junction 5 is the difference of the two flows at the junction.

Table 11.2: A grid network of $m \times m$ nodes: one sparse factorization against one rank-one update, and ten readings folded in by refactoring each time against factoring once and updating ten times. Means agree to 10^{-13}.

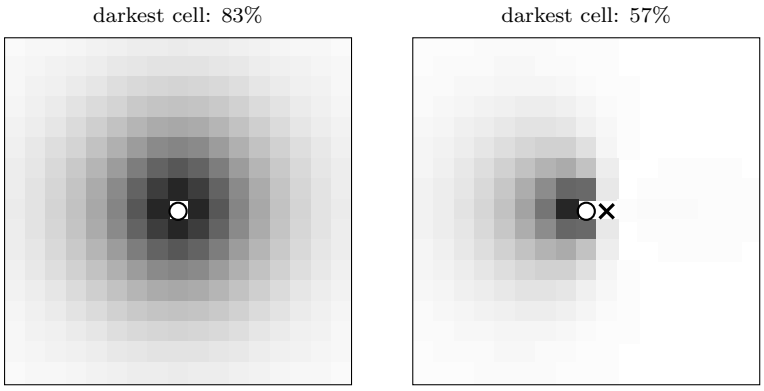
m	unknowns	factor nonzeros	one factorization	one update	ratio	ten readings, ratio
30	900	7.0×10^4	6.2 ms	0.13 ms	48	6.9
100	10,000	2.0×10^6	117 ms	3.5 ms	33	10.7
300	90,000	3.3×10^7	3.55 s	49 ms	72	9.1

sparse structure grows slowly with size. Over ten readings the factor-once route is about ten times faster than refactoring, the first factorization being paid either way; over a hundred it would be some fifty times faster, and the limit is the per-update ratio. The means agree to thirteen decimals. At ninety thousand unknowns a reading is folded in fifty milliseconds, which is faster than sensors report.

11.9 Worked example: where to put the next pressure logger

A dense residential grid is the easiest network in the book to draw and the hardest to instrument: streets on a square lattice, a pipe along every street, a service connection at every junction, and almost nowhere to put a gauge except a hydrant. Model it as the grid of §11.8 at 30×30 junctions with 4 kg/s per metre between neighbouring junctions, 0.25 kg/s per metre from each junction out through its service connections, and an uncertain demand of standard deviation 1 kg/s at every junction. The junction that matters sits at the centre, where the utility has a customer complaining of low pressure and no fitting it can log from, and the head there is the target. Loggers of accuracy 0.05 m can go on any other junction. Under the prior the centre's head has a standard deviation of 0.288 m, and it is correlated with each neighbour at 0.92.

Figure 11.5(a) maps the value of a logger at each junction near the centre, by equation (11.3). The whole map needs one column of the covariance, the centre's, and its diagonal. A neighbour removes 82.6 percent of the centre's variance, a junction two streets away 74.0, three streets 59.1, and seven streets 23.3. The grid spreads the correlation over several junctions, so the value falls



(a) under the prior (b) after the first thermometer

Figure 11.5: The value of a pressure logger at each junction near the centre of a 30×30 grid network, for the head at the centre (ringed), where no logger can go. Darker cells remove more of the centre's current variance; each panel is scaled to its own largest value. (a) Under the prior: the four neighbours are worth 82.6 percent each, and the value falls off slowly with distance. (b) After a logger on the east neighbour (cross), whose own cell is now worth nothing: the opposite neighbour is now worth most, 56.6 percent of what remains.

off slowly with distance — which is the good news for a utility that cannot log where it wants to, and the reason logging a street away is very nearly as good as logging the spot itself.

Panel (b) is the map after that reading. The best next position is the opposite neighbour, now worth 56.6 percent of what remains. The junctions beside the first logger have lost most of their value, and those on the far side have kept it. The first reading told the model the head on one side of the centre, and what remains uncertain is the gradient across it, which the opposite side measures. The second logger brings the centre to 0.079 m. The third, on a side neighbour, removes 14.5 percent of what remains, and the fourth, on the last neighbour, 20.5 percent, more than the third did. A logger's value can rise as others are placed: readings on opposite sides of a junction are worth more together than apart, the explaining away of Proposition 5.2 again. A crew told to put four loggers out and spread them evenly would have done worse than a crew told to surround the complaint.

The four loggers leave the centre at 0.065 m, and no logger outside the centre can do much better. With its four neighbours known exactly the centre would still be uncertain by 0.060 m, and with every other junction of the grid known exactly, by 0.055. What remains is the centre's own demand. It enters only the centre's balance, where it is divided by the total conductance at the junction, $4 \times 4 + 0.25$ kg/s per metre, which sets the scale: $1/16.25$ is 0.062 m. The neighbours' balances recover a little more, through the pipes between them and the centre, and that is all there is. A virtual sensor at the centre is limited not by where the loggers go but by what the physics lets through, and the utility that wanted to know the head at one customer's connection to a centimetre was never going to get it from the street.

11.10 In practice

Hold one factorization and a correction list. Factor the precision once, keep the pairs $(\mathbf{k}_i, \mathbf{s}_i)$ of the updates, and refactor when the list costs more to apply than a factorization costs

to redo. Table 11.2 puts the break-even between thirty and seventy updates for the grid.

Test every reading before folding it in. Compute the standardized innovation from the solve the update needs anyway, and hold aside a reading beyond three spreads. Count the holds per sensor. A sensor that is held once may have glitched; a sensor held repeatedly is drifting, as the meter on pipe 4–5 was.

Run a cumulative test as well, and know what it dilutes. The running sum of squared innovations catches a slow drift that no single reading reveals, and it blurs a sudden fault over every reading that follows. In the stream of §11.5 it dipped back below its quantile between the two biased readings. Use both tests.

Re-linearize by shift, not by surprise. On a nonlinear model the error of the rank-one update grows with how far the reading moves the state, measured in prior standard deviations along the reading. A modest innovation does not certify a small error. One Gauss–Newton step from the updated mean restores the mode.

Choose the next sensor for the question. Compute the value of each candidate for the target that matters, by equation (11.3), and recompute after every reading, since the values change. The whole-state measure of equation (11.4) is for when there is no single target.

Order does not change the answer; it changes what is known early. Every order ends at the batch posterior. When decisions are made between readings, read first the sensor worth most now.

11.11 What is missing

Part III has organized the computation: exactly by regions, approximately where the structure runs out, costed against sampling in solves, and updated one reading at a time without re-solving. What it has not done is ask a question. Every computation so far returned a posterior, and a posterior is not an answer until someone asks it something: what is the head where there is no sensor, is this reading a fault or a leak, what would happen if that pipe were shut, which input drives the risk, which pipes are most likely to fail together. Those are the operator's questions, and Part IV asks them in turn, beginning with the first.

Notes and further reading

The Sherman–Morrison and Woodbury identities and their history are reviewed by Hager [31]. The update (11.1) is the measurement update of the Kalman filter [38], whose gain and innovation are the same objects; Särkkä and Svensson [82] give the modern treatment, including the information form used here and the relation to Gaussian inference on a graph. Computing the covariance column by a solve against a cached sparse factorization rather than by forming the covariance is selected inversion again [21]; the correction-list scheme of §11.2 is the standard way to apply low-rank updates to a factorized matrix without refactoring, and is what a Kalman filter on a sparse state amounts to.

That the covariance reduction from a Gaussian measurement is independent of the measurement's value, and can therefore be used to choose measurements before they are made, is the basis of Bayesian experimental design. Lindley [52] defined the value of an experiment as its expected information; Chaloner and Verdinelli [13] review the Bayesian theory, and the greedy variance-reduction criterion of §11.7 is its simplest instance. The reading of the criterion on the chain, that the room sensor's value goes from nothing to nearly everything once one head is known, is the book's illustration of a general fact: a measurement's value is a property of the current posterior, not of the sensor.

The prediction-error decomposition of equation (11.2) is standard in filtering, where it is how the likelihood of a model's parameters is computed from a run of the filter; Särkkä and Svensson [82] derive it in that setting. The counterexample to greedy placement in the solution to Exercise 11.4 is the explaining-away of Proposition 5.2 read as a design problem.

Summary

- A reading adds a rank-one term to the precision. Conditioning on it is one solve against the held factorization for the covariance column, a gain, and a subtraction never formed. Exact for a linear-Gaussian branch.
- The same solve gives the innovation's predicted spread; the standardized innovation is the running adequacy test. Both of the chain's readings surprised the model by a fifth of a spread.
- Readings in sequence commute; blocks use Woodbury; the whole is the static Kalman filter on a sparse state. For a Laplace posterior, update first and re-linearize only when the innovation or the shift is large.
- The variance a reading removes from a target is known before the reading: squared covariance over total spread. On the chain the trunk main sensor is worth nothing before h_2 is read and 94 percent of the remaining variance after; on the network the meter nearest the target is worth most.
- On a grid of ninety thousand unknowns one update costs a seventieth of a factorization and takes fifty milliseconds.
- Standardized innovations of an adequate model are independent standard Gaussians, and their log densities sum to the evidence. On the network stream a meter drifting by five times its accuracy was flagged at 4.6 spreads on its first biased reading; folding it in would have moved the demand at junction 5 by 1.7 standard deviations the wrong way.
- On a nonlinear pipe the rank-one update errs by two and a half posterior standard deviations when a reading shifts the state by one prior one. One Gauss–Newton step repairs it.
- On a grid network the best place for a gauge near an inaccessible hot spot is a neighbour, and after it the opposite neighbour. Four gauges bring the hot spot to 0.065 metres, against a floor of 0.055 that no outside gauge can pass.
- A sensor's value depends on the target and on what has been read: after the meter on pipe 4–5, the meter on the link 5–6 goes from 44 to 95 percent of the demand's remaining variance.

Exercises

Exercise 11.1. Derive equation (11.1) from the information-form update $\boldsymbol{\Lambda}' = \boldsymbol{\Lambda} + w\mathbf{h}\mathbf{h}^{\mathsf{T}}$, $\boldsymbol{\eta}' = \boldsymbol{\eta} + wy\mathbf{h}$ and the Sherman–Morrison identity. Then derive the Woodbury form for r simultaneous readings and show it reduces to r sequential applications of the rank-one form.

Exercise 11.2. Show that the standardized innovations of a sequence of readings on an adequate linear-Gaussian model are independent standard Gaussians, so that their sum of squares over r readings is chi-squared with r degrees of freedom. Relate this to the misfit statistic of §4.5.

Exercise 11.3. On the chain, read h_a first (say 19.5 with accuracy 0.5), then h_2, then h_1, and confirm that the final posterior equals the batch posterior with all three readings. Then compute the value of each of the three sensors under the prior and after each reading, and show the order in which greedy placement would choose them.

Exercise 11.4. For the two-district network, choose two flow meters greedily to minimize the posterior variance of q_5, then choose the best pair by exhaustive search over the fifteen pairs, and compare. Repeat with the target changed to the transfer flow m_{34}.

Exercise 11.5. Rerun the grid timing with a hundred readings instead of ten, and with the correction list refreshed by a refactorization every r readings for $r \in \{10, 30, 100\}$. Find the r that minimizes total time at $m = 100$ and relate it to the per-update ratio of Table 11.2.

Solutions to selected exercises

Exercise 11.1. The rank-one form is the proof of Proposition 11.1. For r readings at once, stack their rows into $\mathbf{H}$, $r \times n$, with noise covariance $\mathbf{R}$, and write $\mathbf{S} = \mathbf{R} + \mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^{\mathsf{T}}$. The claim is that $\boldsymbol{\Sigma}' = \boldsymbol{\Sigma} - \boldsymbol{\Sigma}\mathbf{H}^{\mathsf{T}}\mathbf{S}^{-1}\mathbf{H}\boldsymbol{\Sigma}$ inverts $\boldsymbol{\Lambda}' = \boldsymbol{\Lambda} + \mathbf{H}^{\mathsf{T}}\mathbf{R}^{-1}\mathbf{H}$. Multiplying,

$$\begin{aligned}\boldsymbol{\Lambda}'\boldsymbol{\Sigma}' &= \mathbf{I} + \mathbf{H}^{\mathsf{T}}\mathbf{R}^{-1}\mathbf{H}\boldsymbol{\Sigma} - \mathbf{H}^{\mathsf{T}}\mathbf{S}^{-1}\mathbf{H}\boldsymbol{\Sigma} - \mathbf{H}^{\mathsf{T}}\mathbf{R}^{-1}\mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^{\mathsf{T}}\mathbf{S}^{-1}\mathbf{H}\boldsymbol{\Sigma} \\ &= \mathbf{I} + \mathbf{H}^{\mathsf{T}}\mathbf{R}^{-1}(\mathbf{S} - \mathbf{R} - \mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^{\mathsf{T}})\mathbf{S}^{-1}\mathbf{H}\boldsymbol{\Sigma} \;=\; \mathbf{I},\end{aligned}$$

where the second line inserts $\mathbf{S}\mathbf{S}^{-1}$ into the second term of the first and $\mathbf{R}^{-1}\mathbf{R}$ into the third. The mean follows as in the rank-one case: $\boldsymbol{\mu}' = \boldsymbol{\mu} + \boldsymbol{\Sigma}\mathbf{H}^{\mathsf{T}}\mathbf{S}^{-1}(\mathbf{y} - \mathbf{H}\boldsymbol{\mu})$. The cost is r solves for $\boldsymbol{\Sigma}\mathbf{H}^{\mathsf{T}}$ and one dense $r \times r$ factorization. To see that r rank-one updates give the same result when $\mathbf{R}$ is diagonal, look at the information form. The batch posterior has precision $\boldsymbol{\Lambda} + \sum_i w_i \mathbf{h}_i \mathbf{h}_i^{\mathsf{T}}$ and information $\boldsymbol{\eta} + \sum_i w_i y_i \mathbf{h}_i$. Each rank-one update is exact, so it produces the posterior whose precision and information have exactly one more term of each sum. After r updates both sums are complete, in whatever order they were taken, and a Gaussian is determined by its precision and information. The stream of §11.5 confirms it to machine precision: its sequential evidence equals the batch predictive density.

Exercise 11.3. In every order the source ends at 104.97 ± 0.97 and the trunk main at 19.52 ± 0.48, the batch posterior with all three rows; Figure 11.1 draws the three paths. The values for the source, as percentages of its current variance, are as follows. Under the prior, h_1 is worth 95.5, h_2 91.5 and h_a nothing, so greedy placement takes h_1. After it, h_a is worth 94.4 and h_2 54.5, so greedy takes h_a. After both, h_2 removes 6.1 percent. The greedy order is h_1, h_a, h_2, and the sensor worth nothing at first is chosen second, because the first reading made the trunk main the largest remaining uncertainty about the source. In the order the exercise prescribes, h_a first, the source's variance is unchanged by the first reading, as its value of zero said it would be.

Exercise 11.4. For the demand at junction 5, greedy placement takes the meter on pipe 4–5 first, worth 64 percent, and then the meter on the link 5–6, which leaves a standard deviation of 0.280 kg/s against a prior of 2.0. Exhaustive search over the fifteen pairs finds the same pair best; the next are 4–6 with 5–6 at 0.356 and 4–5 with 4–6 at 0.736. For the transfer flow m_{34} greedy takes the meter on the transfer main itself and then any other: every pair containing it leaves between 0.199 and 0.200 kg/s, the meter's own accuracy, and the second meter hardly matters. On this network greedy placement of two meters is optimal for the targets tried.

It is not optimal in general, and the failure is explaining away. Let the target be $t \sim \mathcal{N}(0,1)$ and let a nuisance $u \sim \mathcal{N}(0,1)$ be independent of it. Offer three sensors: A reads $t+u$ and B reads u, both with accuracy 0.01, and C reads t with accuracy 0.5. Alone, C removes $1-1/(1+4) = 80$ percent of the target's variance, A removes $1/(2+10^{-4})$, about half, and B nothing, since u is independent of t. Greedy takes C, leaving a variance of 0.2. Then A has covariance 0.2 with the target and variance 1.2, and removes 0.04/1.2, leaving 0.167, while B is still worth nothing. The pair A and B, which greedy never considers, leaves about 2×10^{-4}: their difference reads t to within 0.014. A meter on a flow that carries none of the target can be worthless alone and decisive in a pair. When candidates come in such pairs, search over pairs.

Part IV

The operator's questions

Chapter 12

Conditioning, diagnosis, and virtual sensors

A posterior is not an answer. It is a distribution over every variable of the system, and an operator asks it something specific: what is the head at the node that has no sensor, is this reading a fault or a leak, can this parameter be recovered from these sensors at all, and which sensor should be added next. This chapter shows that each of those questions is a readout of the one posterior, computed from the factorization already held, and that the posterior can also be asked the question that matters most before any other: whether it should be believed. Adequacy, localization, hypothesis comparison, identifiability, experiment design and the validity of the Gaussian approximation are each one section, each with a number on the chain, and each is a solve or a determinant against $\mathbf{\Lambda}$.

12.1 Everything is a readout

Table 12.1 lists the questions of this chapter and the next three against what each reads from the posterior. The point of the table is its right-hand column: nothing in it is a new model. Adding a question does not add a computation to the model; it adds a readout of a posterior that was computed once. That is the practical consequence of building the probabilistic model on the physics' own rows, and it is why the book insisted in Chapter 3 that sensors and priors be added to the graph rather than the graph be built around them.

12.2 Virtual sensors

The marginal of a variable that no sensor reads is a *virtual sensor*: the physics and the sensors that do exist determine it, and the posterior says how well. It is the same marginal as for a measured variable, read the same way; the only difference is whether a likelihood factor happens to touch it. Table 12.2 gives the full posterior of the chain with both head readings, from the script `ch12_diagnosis.py` in the book's `scripts` directory, which supplies every number in the chapter. The two measured heads are known to half a metre, their sensors' accuracy less a little for the physics' corroboration. The two flows, which no instrument reads, are known to 2.9 kg/s, and the source and the room likewise, all from two gauges and four rows of physics.

A virtual sensor is only as good as the model behind it, which is why the rest of the chapter is about testing the model.

Table 12.1: Operational questions as readouts of one posterior.

	Question	Readout
Q1	estimate a measured quantity	its marginal mean and spread
Q2	estimate an unmeasured one (virtual sensor)	the same marginal, on a variable with no sensor
Q3	is the model adequate?	the misfit against its chi-squared law
Q4	where is it inadequate?	studentized residuals, ranked
Q5	which explanation?	log-evidence of each hypothesis
Q6	can this parameter be recovered?	the parameter block's Schur complement and its flattest direction
Q7	which sensor next?	expected information gain, one solve per candidate
Q8	should the Gaussian be believed?	curvature probes and reweighting efficiency
Q9	what if? (Chapter 13)	clamp, re-infer
Q10	which input drives the risk? (Chapter 14)	gradients from the solve

Table 12.2: The chain's posterior with both readings: every variable, with whether a sensor reads it.

variable	posterior	sensor
h_1	92.97 ± 0.47	yes
h_2	72.04 ± 0.44	yes
m_{12}	104.63 ± 2.86	no
m_{2a}	104.63 ± 2.86	no
G	104.63 ± 2.86	no
h_a	19.73 ± 1.73	no

12.3 Is the model adequate?

Chapter 4 named the misfit, the sum of squared standardized residuals over every factor at the posterior mode, and stated its law with its assumptions. Restated:

Proposition 12.1 (Adequacy)**.** *For a linear-Gaussian model with m_g factors (physics rows, priors and sensors, each a Gaussian pseudo-observation) and n unknowns, if the model is correctly specified, the misfit $\chi^2 = \mathbf{g}(\mathbf{z}^\star)^\mathsf{T}\mathbf{\Omega}\,\mathbf{g}(\mathbf{z}^\star)$ is chi-squared distributed with $m_g - n$ degrees of freedom. Its expected value is $m_g - n$, and the probability of a misfit at least as large as the one observed is the model's p-value.*

Proof. Scale each factor by the square root of its weight, so that the factors read $\tilde{\mathbf{c}} = \tilde{\mathbf{H}}\mathbf{z} + \tilde{\mathbf{e}}$, with $\tilde{\mathbf{e}}$ standard Gaussian in m_g dimensions when the model is correctly specified. The mode is the least-squares solution $\mathbf{z}^\star = (\tilde{\mathbf{H}}^\mathsf{T}\tilde{\mathbf{H}})^{-1}\tilde{\mathbf{H}}^\mathsf{T}\tilde{\mathbf{c}}$, and the standardized residual vector is $\tilde{\mathbf{c}} - \tilde{\mathbf{H}}\mathbf{z}^\star = (\mathbf{I} - \mathbf{P})\tilde{\mathbf{c}} = (\mathbf{I} - \mathbf{P})\tilde{\mathbf{e}}$, where $\mathbf{P} = \tilde{\mathbf{H}}(\tilde{\mathbf{H}}^\mathsf{T}\tilde{\mathbf{H}})^{-1}\tilde{\mathbf{H}}^\mathsf{T}$ is the orthogonal projection onto the columns of $\tilde{\mathbf{H}}$. The second equality holds because $(\mathbf{I} - \mathbf{P})\tilde{\mathbf{H}} = \mathbf{0}$. The misfit is $\tilde{\mathbf{e}}^\mathsf{T}(\mathbf{I} - \mathbf{P})\tilde{\mathbf{e}}$, and $\mathbf{I} - \mathbf{P}$ is a symmetric projection of rank $m_g - n$. In a basis of its eigenvectors the misfit is the sum of squares of $m_g - n$ independent standard Gaussians, which is the chi-squared law. □

A prior counts as a pseudo-observation of its variable at its mean, and the proof treats it as one; that is where "correctly specified" bites. A broad prior that only keeps a variable

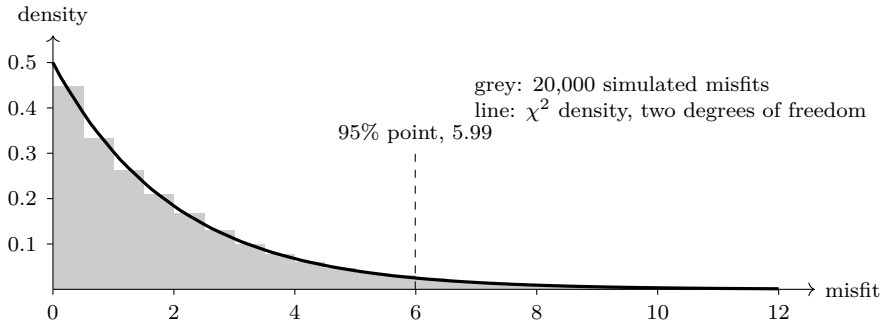


Figure 12.1: Twenty thousand misfits of the chain's model on readings generated from the model itself, against the chi-squared density with two degrees of freedom that Proposition 12.1 predicts. The dashed line is the 95 percent point, above which 4.9 percent of the simulated misfits fall.

identifiable, and from which the truth is not drawn, is not a pseudo-observation in this sense. It belongs in neither the misfit nor the count, and a companion diagnosis study in the power domain, committed to the repository that accompanies the book, measures what counting it costs on a real network. Chapter 22 shows the other side of the same coin on a water network: a misfit that passes because the model was given enough freedom to absorb the error, rather than because the model was right. The script `ch12_more.py` checks the law by simulation. Twenty thousand reading pairs generated from the chain's own priors and physics give a mean misfit of 2.00 against the law's 2, a variance of 3.89 against 4, and 4.9 percent of misfits above the 95 percent point. The physics in those draws was exact, not perturbed at its stated width of 0.01 kg/s. Perturbing it at that width changes the three numbers to 2.01, 4.12 and 5.2 percent, the same law within simulation error. Figure 12.1 draws the simulated misfits against the density the proposition predicts.

When $m_g - n \leq 0$ the model has no redundancy and adequacy cannot be assessed; a model that exactly fits its data by construction has not been tested by it, and the honest report is that verdict rather than a comfortable fit. On the chain with two readings there are eight factors and six unknowns: two degrees of freedom.

With the consistent readings 72.0 and 93.0 the misfit is 0.07 against an expectation of 2; the p-value is 0.97. With the inconsistent pair 72.0 and 105.0 of Chapter 4, and the physics rows at a diagnostic width of one kg/s so that they may be violated, the misfit is 70 and the p-value is 5×10^{-16}. The model is wrong, and the test says so with no knowledge of how.

12.4 Where is it wrong?

The natural next step is to look at the residuals one factor at a time. The raw residual of a sensor, $y - \mathbb{E}[z]$, has a variance smaller than the sensor's, because the fitted mean was pulled toward the reading; the right statistic divides by $\sqrt{\sigma_y^2 - \mathrm{Var}[z]}$, the *studentized* residual, and the same correction applies to any factor. This is the classical bad-data statistic of state estimation and the gross-error statistic of data reconciliation, and it is one selected inversion per factor. The variance $\sigma_y^2 - \mathrm{Var}[z]$ is derived in the solution to Exercise 12.1.

For the inconsistent readings the studentized residuals are, in order of the factors: balance at node 1, $+2.4$; balance at node 2, -7.8; flow closure, -8.4; the valve closure, $+7.8$; source

Table 12.3: Redundancy of each meter on the two-district network, and the smallest gross error that produces a studentized residual of three. Every meter has accuracy 0.2 kg/s.

meter	redundancy r_k	smallest error flagged [kg/s]
m_{R2}, m_{23}	0.77, 0.75	0.68, 0.70
m_{45}, m_{56}, m_{46}	0.63–0.52	0.75–0.84
d_2, d_5	0.34, 0.35	1.03, 1.01
m_{R3} (the large main)	0.15	1.53
m_{34} (the transfer main)	0.019	4.32

prior, +2.4; room prior, −7.8; the two sensors, +8.3 and −8.4. Six of the eight factors are violated by eight standard deviations, and the pattern does not point at one of them. It cannot. The two readings imply a source of 165 kg/s and a room of −10 metres, and a model with tight physics distributes that contradiction across every factor that could absorb a piece of it. Ranking residuals localizes a fault when one factor is far more violated than the rest; when the contradiction is global, as here, it says only that something is wrong everywhere downstream of the readings.

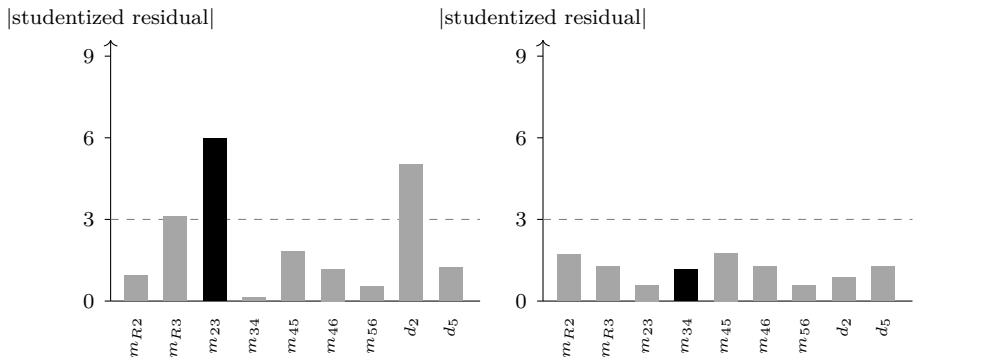
Principle 12.1. *Studentized residuals localize a fault when it is local. A global contradiction spreads across every factor that can absorb it, and ranking residuals then says where the model bends, not what broke.*

12.5 Worked example: bad data on a water network

Localization works when the fault is local and the model has redundancy around it. The two-district network shows both conditions, and what happens when the second fails. Give it a flow meter on each of its seven pipes and a demand meter at junctions 2 and 5, all of accuracy 0.2 kg/s, with the demand priors of Chapter 8. Twenty-six factors on seventeen unknowns leave the misfit nine degrees of freedom. The readings are the network's own state plus one draw of each meter's noise; with them the misfit is 6.74, a p-value of 0.66, and no studentized residual exceeds 1.83. Every number is from `ch12_baddata.py`.

A meter's *redundancy* is the fraction of its variance that the rest of the model leaves unexplained, $r_k = 1 - \mathrm{Var}[\mathbf{h}_k^\mathsf{T}\mathbf{z}]/\sigma_k^2$, with the variance taken under the posterior. A gross error b on meter k moves its residual by $b\,r_k$, and the studentized residual divides by $\sigma_k\sqrt{r_k}$, so the error shows up as $b\sqrt{r_k}/\sigma_k$. The smallest error flagged at three spreads is therefore $3\sigma_k/\sqrt{r_k}$. Table 12.3 lists both. The meters on the pipes inside the two loops have redundancy between 0.52 and 0.77, and an error of three quarters of a kilogram a second on any of them is caught. The meter on the main R–3 has 0.15: it carries two thirds of the district's supply, and the only things that check it are the other main and the five demand priors, which are ten times less precise. The meter on the transfer main has 0.019. An error there must reach 4.32 kg/s, twenty-two times the meter's accuracy and a fifth of the flow it is measuring, before the test flags it.

The reason is the graph. The transfer main is a bridge, the only pipe between the districts, so nothing else in the model measures the flow it carries except the demand priors. A meter on a bridge is what state estimation calls a critical measurement. It is also the port of Chapter 8 seen from the other side: a separator of size one, across which the two districts can only be reconciled



(a) meter on link 2–3 reads 1.5 kg/s high (b) meter on the transfer main reads 1.5 kg/s high

Figure 12.2: Studentized residuals on the two-district network when one meter reads 1.5 kg/s high, seven and a half times its accuracy. Black marks the faulty meter, and the dashed line is at three. (a) A meter inside a loop: the fault leads the ranking. (b) The meter on the transfer main, a bridge: no residual reaches two, the faulty meter does not lead, and the error is absorbed into the demands at the two ends of the pipe.

through the one number the meter reads. And the reconciliation is mass, not pressure: whatever the heads do, the transfer main carries the demand beyond it, which is precisely why no other reading can contradict a reading of it.

Now make one meter read 1.5 kg/s high, seven and a half of its accuracies. Figure 12.2 draws the studentized residuals. With the fault on the meter on link 2–3, the misfit jumps to 42.2, a p-value of 3×10^{-6}, and the ranking puts the faulty meter first at 5.97, ahead of the demand meter at junction 2 at 5.01 and the meter on the main R–3 at 3.11. The meters around a fault share its residual, but the fault itself leads. With the fault on the transfer main, the misfit is 8.07, a p-value of 0.53, and no residual exceeds 1.77; the faulty meter is not even among the four largest. The test sees nothing. Meanwhile the estimate of the transfer flow has moved 7.4 posterior standard deviations and the head at junction 4 the same amount the other way, and the demands at junctions 3 and 4 have moved 4.6 and 4.5 in opposite directions. The model has absorbed the error by trading demand between the two ends of the bridge, which is the only explanation the graph allows, and no residual can contradict it.

Principle 12.2. *A measurement is only as checkable as it is redundant. A meter on a bridge of the network, a separator of size one, has nothing to be checked against, and a gross error on it moves the state without moving the residuals. Redundancy is computed before any reading, from the same diagonal as the virtual sensors.*

12.6 Which explanation?

When the residuals cannot say what broke, name the candidates and let the evidence choose. Each candidate is the model plus one latent variable with a zero-centered prior: a leak at node 2, which adds a loss term to that balance; a leak at node 1; a bias on the h_1 sensor, which adds an offset to its likelihood; a bias on the h_2 sensor. Each is a linear-Gaussian model with one more variable, and each has an evidence, the integral of its factors over all its variables, which

Table 12.4: Five explanations of the inconsistent readings 72.0 and 105.0, scored by log-evidence. Posterior probabilities at equal prior odds. Physics width 0.01 kg/s.

hypothesis	fault prior	$\log Z$	G [W]	fault estimate	probability
nothing wrong	–	-43.8	147	–	0.000
leak at node 2	50 W	-14.6	163	57 ± 7 W	0.010
leak at node 1	50 W	-42.4	107	-41 ± 19 W	0.000
bias on h_1 sensor	5 K	-10.9	107	11.3 ± 1.4 K	0.387
bias on h_2 sensor	5 K	-10.5	122	-8.5 ± 1.0 K	0.603

Chapter 7 gave in closed form:

$$\log Z \;=\; -C(\mathbf{z}^\star) - \tfrac{1}{2}\log\det\boldsymbol{\Lambda} + \tfrac{d}{2}\log 2\pi + \sum_k \log\frac{1}{\sqrt{2\pi}\,\sigma_k}, \tag{12.1}$$

with d the number of variables and the last sum over every factor's normalizer. The constants matter here because the hypotheses have different numbers of variables. One factor that does not matter here is the determinant of the physics rows' Jacobian with respect to the solved variables, which Chapter 7 had to restore when its branches changed a closure (Proposition 7.1): every hypothesis in this section adds its latent variable to a balance or a sensor row as an input with a prior, and leaves the physics Jacobian as it was, so that factor is common to all of them and cancels. The determinant term that remains is the *Occam factor*: it charges each hypothesis for the volume of states its extra variable makes plausible, so that a hypothesis with a wide prior on its fault variable is not rewarded merely for being able to fit anything.

Table 12.4 gives the result, and it is not the one Chapter 4 assumed. The null hypothesis and the leak at node 1 are ruled out by thirty nats. The leak at node 2, which Chapter 4 loosened a balance to find, is a possible explanation, at one percent. The two sensor biases are the probable ones, sixty percent to thirty-nine, an eleven-degree bias on one gauge or an eight-degree bias on the other. The reason is in the columns: the leak explanation needs a leak of 57 kg/s, comfortable under its 50-kg/s prior, but also a source of 163 kg/s, three spreads above its nominal 100; the bias explanations need a bias of two spreads and a source within one. The evidence weighs every factor, and the source prior tips it.

Two things follow. First, the verdict depends on the priors given to the fault variables, and it should: a modeler who knows the gauges are freshly calibrated would put a one-degree prior on the biases, and the leak would win. The computation makes that judgment explicit and prices it. Second, the computation is one factorization per hypothesis, each on a model one variable larger than the base, and the hypotheses can be as many as the operator can name. Chapter 15 uses the same evidence to rank hundreds of them.

Figure 12.3 prices the first judgment. It holds the leak's prior at 50 kg/s and varies the width of the bias priors. Below 2.3 metres the leak at node 2 is the better explanation. Above it the bias on h_2 is, and above 6.3 metres the bias on h_1 takes over. The evidence for a bias rises steeply as its prior widens from half a metre, because a narrow prior cannot reach the eight- or eleven-degree bias the readings need, and the fit pays for the shortfall. It peaks near ten metres, at 9 for the bias on h_2 and 12 for the bias on h_1, and then falls slowly. A prior far wider than needed spends its probability on biases that did not happen, which is the Occam factor at work.

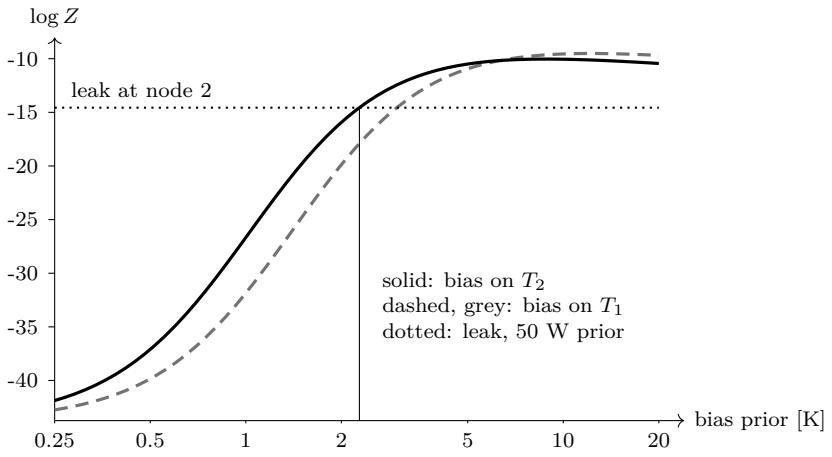


Figure 12.3: Log-evidence of the fault hypotheses for the inconsistent chain readings, as the width of the sensor-bias priors varies with the leak's prior held at 50 kg/s. The thin vertical line marks the crossover at 2.3 metres, below which the leak is the better explanation.

12.7 Can this parameter be recovered?

Whether a parameter can be recovered from the sensors at hand is a property of $\mathbf{\Lambda}$ alone, before any reading. Partition the variables into the parameters of interest $\mathbf{p}$ and everything else, and take the Schur complement onto the parameters as in Chapter 8. By block elimination the $(\mathbf{p}, \mathbf{p})$ block of $\mathbf{\Lambda}^{-1}$ is the inverse of $\mathbf{\Lambda_{pp}} - \mathbf{\Lambda_{ps}}\mathbf{\Lambda_{ss}}^{-1}\mathbf{\Lambda_{sp}}$, so the Schur complement's inverse is the marginal parameter covariance. Its eigenvectors are the directions in parameter space the data constrain, and the eigenvector of the smallest eigenvalue is the *flattest* direction, the combination of parameters the data cannot separate. A scale-free summary per parameter is the ratio of its posterior spread to its prior spread, ρ_i: near zero, the data pinned it; near one, the data said nothing.

Free the conductance k on the chain with a prior so wide as to be no prior, and ask what the sensors can do for it. With only the h_2 reading, $\rho_k = 0.999$ and the flattest direction is k itself: a single head two nodes from the conductance says nothing about it, because h_2 depends on the source and the trunk main and not on how the source's heat gets to the junction. Add the h_1 reading and ρ_k falls to zero, with k known to ± 0.34 kg/s per metre: the head drop across the path, $h_1 - h_2$, is the source over the conductance, and the source is known from h_2. The flattest direction is then the source-room ridge, as it has been since Chapter 3. None of this needed a reading; it needed the sensors' positions and the model.

12.8 Worked example: can tuberculation be seen?

An unlined cast-iron main tuberculates: the inside of the pipe grows a crust, the bore narrows and roughens, and the Hazen–Williams coefficient C that Appendix D parameterizes the head loss with falls from around 130 when the main was laid towards 80 or worse. A utility wants to know when to clean or reline it, and it has the cheapest instrument there is on either side of the length: a pressure gauge.

Take 1,200 m of 300 mm main between two gauges. Its roughness coefficient has a prior of

Table 12.5: Can tuberculation be seen? The posterior spread of the length's roughness coefficient for four sensor sets, its ratio to the prior spread, and the shift of the roughness estimate caused by a 15 percent loss of C, in units of its own posterior spread.

sensors	sd of C	ρ_C	the loss shows at
the gauge pair alone	11.08	0.74	1.0 sd
+ a flow meter at 2%	2.54	0.17	9.4 sd
+ a flow meter at 0.5%	0.68	0.05	36.2 sd
+ a pumping-energy meter	2.59	0.17	9.2 sd

130 ± 15 and the flow through it, known only from the demand estimate for the district it feeds, a prior of 95 ± 12 kg/s. The gauges read the head drop to 0.02 m. At the nominal values the length loses 7.013 m and costs 6.5 kW of pumping power. Every number is from `ch12_more.py`.

The trouble is visible in the closure before any number is computed. The head drop is

$$\Delta h \;=\; \frac{10.667\,L}{C^{1.852}D^{4.871}}\,Q^{1.852} \;=\; \frac{10.667\,L}{D^{4.871}}\Big(\frac{Q}{C}\Big)^{1.852}, \tag{12.2}$$

which depends on the roughness and the flow only through their ratio.

Proposition 12.2 (A symmetry the gauges cannot see)**.** *The head drop across a length of main is unchanged when C and the flow are multiplied by a common factor. The Fisher information that the gauges carry about $(C, \dot m)$ therefore has the vector $(C, \dot m)$ in its null space, and no number of gauges, however accurate, can resolve that direction.*

Proof. Under the scaling by λ the ratio Q/C is unchanged, so by equation (12.2) Δh is unchanged. The head drop is therefore constant along the ray $\lambda(C, \dot m)$, its derivative along that ray is zero, and the vector lies in the null space of the gauges' Jacobian $\mathbf{J}$ and hence of the Fisher information $\mathbf{J}^{\mathsf{T}}\mathbf{R}^{-1}\mathbf{J}$. □

The script confirms it. Scaling both by 0.85 or by 1.2 leaves the head drop at 7.012727 m to six decimals, while the pumping power moves to 5.6 or 7.8 kW. The smallest eigenvalue of the gauges' Fisher information is zero to machine precision, and its eigenvector is (0.807, 0.590), which is the scaling ray $(C, \dot m)/\|(C, \dot m)\|$ to three decimals. What the gauges do fix is the ratio, and nothing else.

Now let the main tuberculate by fifteen percent, C from 130 to 110.5, and ask each sensor set whether it noticed. Table 12.5 answers, and Figure 12.4 draws the first two cases.

With the gauge pair alone the roughness is known to 11.1, which is three quarters of what the prior already said, and the fifteen percent loss moves the estimate by *one* posterior standard deviation. It is invisible. The reason is the proposition: the extra head drop is perfectly explained by a slightly larger flow, and the posterior splits the change between the two along the ray it cannot see. Two gauges on a main, which is what most lengths of main have, cannot tell a utility when to clean the pipe.

Add a flow meter of two percent accuracy and the roughness is known to 2.5, a sixth of the prior, and the same loss shows at 9.4 standard deviations. Nothing about the gauges changed. What changed is that the ray is no longer free, so the head drop's information, which was always there, is now about the roughness alone. Improve the flow meter to half a percent and the loss

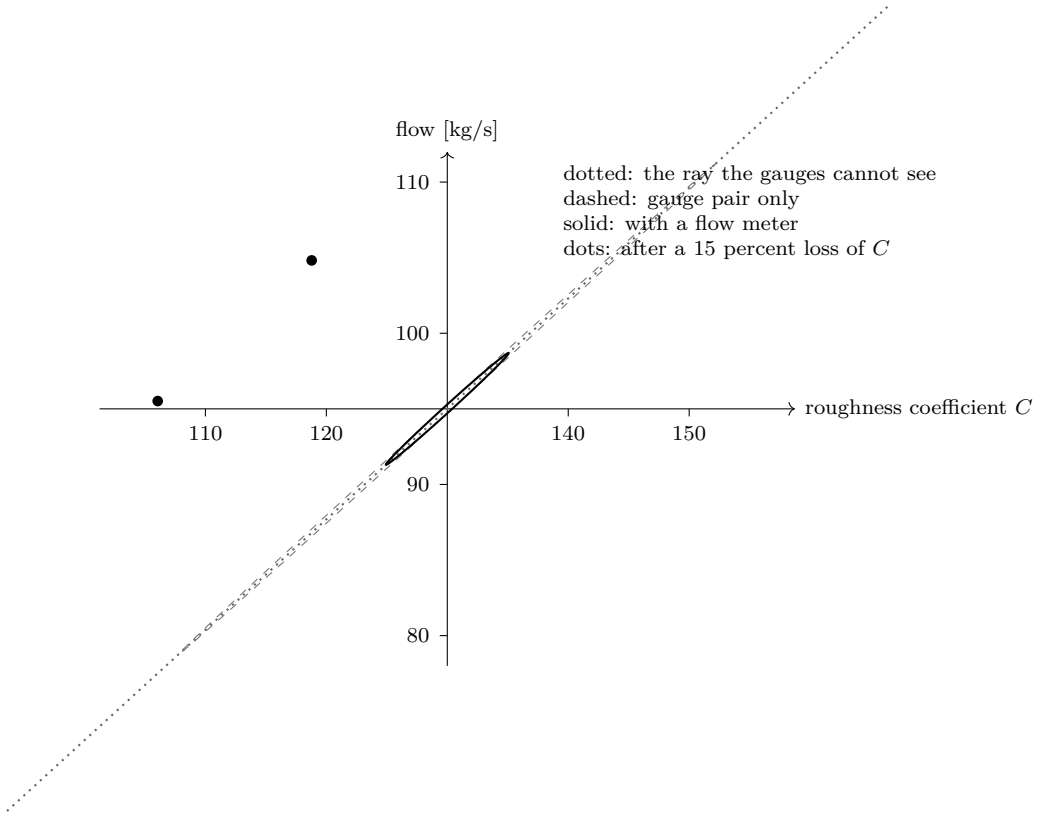


Figure 12.4: The length of main's posterior over its roughness coefficient and its flow, at two standard deviations. Dashed grey: the gauge pair only, an ellipse along the scaling ray (dotted), the direction the gauges cannot see. Solid: with a two-percent flow meter. Dots: the posterior means after a 15 percent loss of C. With gauges only the change is split between the roughness and the flow; with the flow meter it goes to the roughness.

shows at 36 standard deviations: the accuracy of the flow meter, not of the gauges, is what sets how early tuberculation is seen.

The fourth row is the one to think about before buying anything. A pumping-energy meter on the length — which reads $\rho g Q\,\Delta h$ and which a pumping station usually already has — breaks the symmetry too, because power scales with λ where head drop does not, and it brings the roughness to 2.6 and the loss to 9.2 standard deviations, within a hair of the dedicated flow meter. The cheapest new instrument is sometimes one that is already installed for another purpose, and the way to find out is to ask which readings lie off the null direction rather than which readings sound most relevant.

Principle 12.3. *A transformation of the inputs that leaves every reading unchanged is a direction no data can resolve. Find the symmetries of the physics before choosing sensors; each one needs a sensor that breaks it.*

Table 12.6: Four candidate sensors on the chain under the prior: expected information gain about the whole state, and the spread of the source after the reading (the criterion of Chapter 11).

sensor on	accuracy	EIG [nats]	sd of G after
h_1	0.5	3.36	4.25
h_2	0.5	3.04	5.82
h_a	0.5	1.81	20.00
m_{2a}	2.0	2.31	1.99

12.9 Which sensor next?

Chapter 11 chose the next sensor by the variance it would remove from one target. The general criterion is the information it would add about the whole state, and for a Gaussian model it has a closed form that, like the variance reduction, does not depend on the reading:

$$\text{EIG} \;=\; \tfrac{1}{2}\log\det(\mathbf{I}+\boldsymbol{\Sigma}\,\mathbf{H}), \qquad \text{EIG} = \tfrac{1}{2}\log\Big(1+\frac{\mathbf{h}^{\mathsf{T}}\boldsymbol{\Sigma}\mathbf{h}}{\sigma_y^2}\Big) \text{ for one sensor,} \tag{12.3}$$

with $\mathbf{H}$ the Fisher information the candidate contributes. This is the expected reduction in the posterior's entropy, in nats, and choosing the sensor that maximizes it is D-optimal design. For one sensor it is one solve. The one-sensor form is equation (11.4) of Chapter 11; the block form follows the same way, from $\det(\boldsymbol{\Lambda}+\mathbf{H}) = \det\boldsymbol{\Lambda}\,\det(\mathbf{I}+\boldsymbol{\Sigma}\mathbf{H})$.

Table 12.6 sets the two criteria side by side under the prior. They agree on the best single sensor, h_1, and disagree elsewhere: the trunk main gauge carries 1.8 nats about the state, since it pins one of the two uncertain inputs outright, and no information at all about the source, since under the prior the two are independent; the flow meter, poor as it is, carries less information in total but nearly all that there is about the source. Information gain is the right criterion when the whole state matters, and the target criterion when one quantity does, and the posterior supplies both from the same solve. For discrete hypotheses, where the value of a reading depends on which hypothesis it favors, the corresponding quantity is the mutual information between hypothesis and reading, and it is evaluated by quadrature over the predictive mixture of Chapter 7.

12.10 Should the Gaussian be believed?

Every readout above assumed the posterior is Gaussian, exactly for linear rows and by the Laplace approximation otherwise. Chapter 7 named the three ways the approximation fails and promised tests that detect them from the model itself. Two tests, both built from what the mode solve already produced, do it.

Curvature. Along an eigenvector of the posterior covariance, the Laplace posterior predicts that the objective rises by exactly $\frac{1}{2}k^2$ at k standard deviations from the mode, whatever the eigenvalue. Evaluate the true objective there and take the ratio to the prediction: one for a quadratic objective, far from one where a row curves within the posterior's width. Probing the widest directions is enough, since they are where the posterior extends farthest.

Table 12.7: Laplace validity probes on the chain with the natural-the valve closure $Q = C\,\Delta T^{p}$ and one h_2 reading, in input space. Curvature: objective increment over its quadratic prediction along the widest posterior axis. ESS: effective sample size fraction under importance reweighting, 20,000 draws.

p	sensor accuracy	sd of G	curvature at ±1 sd	at ±2 sd	ESS/N
1.25	0.5	7.1	1.01, 0.99	1.03, 0.99	0.995
4	0.5	15.3	1.11, 1.00	1.40, 1.06	0.86
1.25	0.05	7.0	1.16, 1.13	1.67, 1.50	0.59
4	0.05	15.2	6.7, 4.9	29, 15	0.066

Reweighting. Draw from the Laplace Gaussian and weight each draw by the ratio of the true posterior density to the Gaussian's. If the two agree the weights are uniform and the *effective sample size*, $(\sum w)^2 / \sum w^2$, equals the number of draws; where they disagree the weights concentrate and the ratio collapses.

Table 12.7 runs both on the natural-the valve chain of Chapter 7, in the four combinations of a mild and a strong exponent with a coarse and a precise sensor. With the coarse sensor the probes confirm what Chapter 7 found: the Gaussian holds, the curvature ratios are within a few percent of one, and 99 percent of the draws are effective at $p = 1.25$ and 86 percent at $p = 4$. Make the sensor ten times more precise and the picture changes. At $p = 1.25$ the ratios reach 1.7 at two spreads and the effective sample fraction falls to 0.59: a warning. At $p = 4$ the objective at one spread is nearly seven times its quadratic prediction and at two spreads nearly thirty; only seven percent of the draws are effective. The Gaussian is wrong, and the probes say so without any exact posterior to compare against.

The mechanism is the third failure mode of §7.4, now seen. A precise sensor confines the posterior to a thin band around the curve $h_a + (G/C)^{1/p} = 72$ in the input plane. A thin band around a curve is a curved ridge, and a Gaussian, whose ridges are straight, cannot follow it. Along the Gaussian's widest axis, which is straight, the true objective rises several times faster than the quadratic on both sides, because the straight axis leaves the curved ridge in both directions. The posterior spread of G is no narrower than with the coarse sensor, so the width test of Chapter 7 would not have noticed; the curvature test does.

Figure 12.5 draws the ratio as a function of distance along the widest axis for all four cases. The coarse-sensor curves stay within ten percent of one out to one spread, and within forty percent at two. The precise sensor at $p = 1.25$ sits between, at 1.5 and 1.7 at two spreads. The precise sensor at $p = 4$ leaves one at once: 4.9 at one spread on one side and 6.7 on the other, 14.7 and 29 at two. The asymmetry is the ridge's curvature, which bends one way.

Escalation. When the probes fail the answer is not to return the Gaussian anyway. The cheap correction is importance reweighting itself: the reweighted draws are samples from the true posterior, and their weighted moments correct skew and curvature near the mode, at the cost of the draws and with the effective sample size as the measure of how well. It cannot find a mode it never visits. It also inherits the scale of the Gaussian it draws from, so where that scale is itself wrong the reweighting cannot repair it. Genuine multimodality, the first failure mode, escalates to a sampler that does not start from the Laplace Gaussian at all: a particle filter that draws the parameters from the prior and integrates the variables the physics determines out of

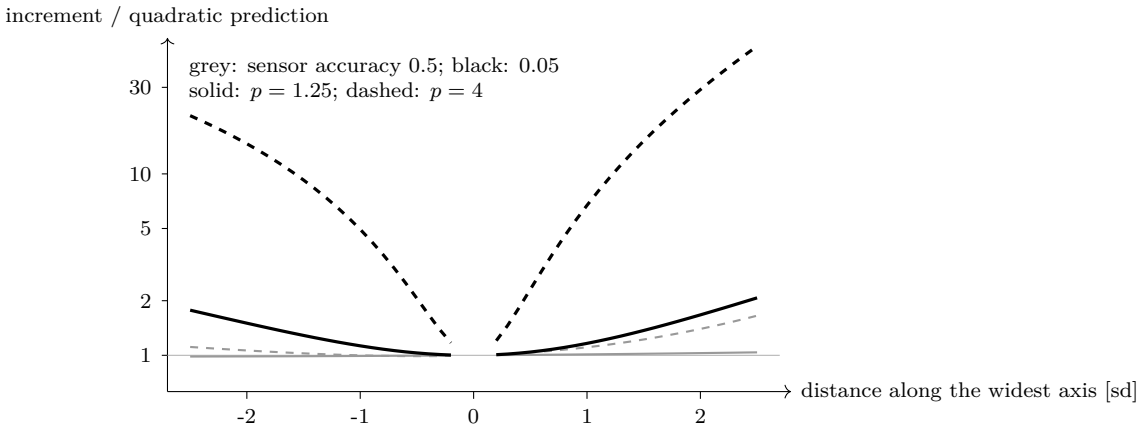


Figure 12.5: The curvature probe on the natural-the valve chain as a function of distance along the widest posterior axis: the increment of the objective over its quadratic prediction, on a logarithmic scale, for the four cases of Table 12.7. A Gaussian posterior would give one everywhere.

each particle, at the cost of one solve per particle. All three return the same readouts, so what changes downstream is only how much they cost.

> **Principle 12.4.** *A posterior is a claim, and the model carries the instruments to test it: the misfit for adequacy, studentized residuals for location, evidence for explanation, the parameter Schur complement for recoverability, and curvature and reweighting for the Gaussian itself. None needs anything the mode solve did not produce.*

12.11 In practice

Run the misfit test first, and report its metres of freedom. A p-value on zero degrees of freedom does not exist, and a model with none has not been tested by its data. Report the verdict together with the count.

Compute every sensor's redundancy before trusting its residual. Redundancy comes from the posterior's diagonal and needs no reading. A meter with redundancy below about a tenth cannot be checked. The fix is another measurement that covers the same quantity: a second meter across the bridge, or a demand meter at one of its ends.

Look for the physics' symmetries before placing sensors. A common scaling of a conductance and the flows through it, a common offset of every head in a model driven by differences, a change of a bridge's resistance that only the heads across it feel: each leaves some set of sensors blind. Test each candidate sensor set against them. The flattest direction of the parameter Schur complement finds the symmetries that nobody thought of.

Rank studentized residuals only when the misfit fails, and expect company. The meters around a local fault share its residual. The fault leads the ranking when it is local and redundant; when the ranking is flat, the contradiction is global.

When residuals smear, name hypotheses and write down their priors. Score each by its evidence. The verdict depends on the fault priors, so report them, and report where the verdict would change, as Figure 12.3 does.

Probe the Gaussian where sensors are precise and closures curve. Evaluate the objective at one and two spreads along the widest axes. A ratio beyond about two, or an effective sample fraction below a half, means the readouts need reweighting or a sampler.

A virtual sensor is valid only while the misfit passes. Put the misfit next to every virtual reading an operator sees. When it fails, the virtual readings are the first casualty.

12.12 What is missing

Every question here was asked of the world as it is. The next two chapters ask about the world as it might be: what would happen if a link were taken out or a setpoint changed (Chapter 13), and which uncertain input the answer depends on most (Chapter 14). The first needs an idea the factor graph does not contain, which is what it means to intervene; the second needs the solve to return its gradient.

Notes and further reading

The misfit test and the studentized residual are the chi-squared test and the normalized residual of power-system state estimation, developed for bad-data detection and identification and treated in full by Monticelli [64] and Abur and Gómez Expósito [1]; the correction $\sigma_y^2 - \mathrm{Var}[z]$ in the denominator is theirs. The same statistics are the global test and the measurement test of process data reconciliation [57]. The observation that a global contradiction spreads across every factor, so that residual ranking fails exactly when it is most needed, is old in both fields under the name of smearing, and hypothesis comparison by evidence is the standard remedy in the reconciliation literature as "serial elimination" of suspected gross errors. Measurement redundancy and critical measurements, those whose removal leaves part of the state unobservable, belong to the observability analysis of state estimation [1, 64]; the meter on the transfer main of §12.5 is the textbook case, and it is the same case whether the bridge carries power or water.

The evidence with its Occam factor, equation (12.1), and its use to compare models of different dimension are MacKay's [56]; Kass and Raftery [41] review Bayes factors. Identifiability through the Fisher information's eigenstructure is classical; the parameter Schur complement is its Gaussian-posterior form. Expected information gain and D-optimal design are Lindley's [52] and are reviewed by Chaloner and Verdinelli [13]. The curvature probe is the book's simplest form of a check on the Laplace approximation; the effective sample size under importance reweighting is standard [76], and the escalation to a particle filter over the prior is Gordon, Salmond and Smith's bootstrap filter [27].

Summary

- Every operational question is a readout of one posterior; adding a question adds no model.

- A virtual sensor is the marginal of an unmeasured variable. Two gauges and four rows of physics give the chain's flows to 2.9 kg/s.
- The misfit is chi-squared with factors minus unknowns metres of freedom, because the standardized residuals are a projection of rank $m_g - n$ of standard Gaussian noise; 0.07 on two for consistent readings, 70 for inconsistent. Studentized residuals localize local faults; a global contradiction smears across every factor.
- A meter on a bridge of the network, the transfer main, has redundancy 0.019. An error of 1.5 kg/s on it passes the misfit test at a p-value of 0.53 and moves the demands at the pipe's ends by 4.6 standard deviations; the same error inside a loop is flagged at 5.97.
- Named hypotheses scored by evidence decide what residuals cannot: for the inconsistent readings, a biased sensor at 99 percent over a leak at one, and the verdict is priced by the fault priors. Below a 2.3-metres bias prior, the leak wins.
- Symmetries of the physics are blind spots of the sensors. Scaling a length of main's roughness and its flow together leaves its head drop unchanged, so a gauge pair alone sees a 15 percent loss of roughness at one standard deviation; a two-percent flow meter makes it nine. No flow meter sees the resistance of a bridge; a differential gauge across it does.
- Recoverability is the parameter Schur complement's flattest direction: k is invisible to one gauge and known to 0.34 with two.
- Expected information gain is a log-determinant, one solve per sensor; it and the target criterion agree on the best sensor and disagree on the trunk main gauge, for a reason the graph explains.
- Curvature probes and reweighting detect a failed Gaussian from the model alone: a precise sensor on a strongly curved closure gives a curvature ratio of 29 and seven percent effective draws.

Exercises

Exercise 12.1. Derive the variance of a sensor's raw residual $y - \mathbb{E}[z]$ under the posterior, $\sigma_y^2 - \mathrm{Var}[z]$, from the rank-one update of Chapter 11, and show it is always positive.

Exercise 12.2. Re-run the hypothesis comparison of Table 12.4 with the sensor-bias priors tightened to one degree. Find the prior width at which the leak at node 2 overtakes the biases, and explain the crossover in terms of equation (12.1)'s terms.

Exercise 12.3. Add a hypothesis to Table 12.4: both sensors biased, each with a five-degree prior. Compute its evidence, split it into the fit term and the rest, and explain how it scores against the single-bias hypotheses.

Exercise 12.4. For the two-district network of Chapter 8 with the transfer main's resistance K_{34} freed at 20 percent, compute the parameter Schur complement onto the five demands and the resistance for each single flow meter, and find the meter for which the resistance has the smallest ρ. Compare with the expected information gain of each meter.

Exercise 12.5. Implement the profiled curvature probe: instead of moving the full state along an eigenvector, clamp one variable at $\pm k$ spreads and re-minimize the rest. Apply it to the $p = 4$, precise-sensor case of Table 12.7 and compare the ratios with the unprofiled ones; explain why the profiled probe stays on the physics manifold and the unprofiled one need not.

Solutions to selected exercises

Exercise 12.1. Let v be the variance of z under the posterior without the sensor, and $S = \sigma_y^2 + v$ the predicted variance of the reading. By Proposition 11.1 the posterior mean with the sensor is $\mu' = \mu + (v/S)(y - \mu)$, so the raw residual is $y - \mu' = (y - \mu)\,\sigma_y^2/S$. Under the model $y - \mu$ has variance S, so the raw residual has variance σ_y^4/S. The posterior variance with the sensor is $\mathrm{Var}[z] = v - v^2/S = v\sigma_y^2/S$, and $\sigma_y^2 - \mathrm{Var}[z] = \sigma_y^2(S - v)/S = \sigma_y^4/S$, the same number. It is positive whenever the sensor has any noise at all. It falls toward zero as v/σ_y^2 grows, which is a sensor whose quantity nothing else in the model knows: the fit follows the reading, the residual cannot move, and a gross error on it is invisible. Divided by σ_y^2 it is the redundancy $r_k = \sigma_y^2/S$ of §12.5.

Exercise 12.2. At a one-degree bias prior the bias on h_2 has $\log Z = -26.7$ against the leak's -14.6, and the leak wins by twelve nats. The crossover is at 2.28 metres (Figure 12.3). In the terms of equation (12.1), the fit term $-C(\mathbf{z}^\star)$ of the bias hypothesis is -19.5 at one degree and -2.1 at five, because a one-degree prior can afford only a 4.3-degree bias, half of the 8.5 the readings need, and the rest of the contradiction is paid by the source prior and the sensors. The volume and normalizer terms together move by much less, from -7.2 at one degree to -8.4 at five. Tightening the prior makes the bias cheaper in Occam terms by 1.3 nats and dearer in fit by 17.4, and the fit wins. The leak's terms do not change.

Exercise 12.3. The two-bias hypothesis has $\log Z = -9.95$, against -10.50 for the bias on h_2 alone and -10.95 for the bias on h_1 alone: it wins, by 0.55 nats over the better single bias, odds of 1.7. Split against the single bias on h_2, its fit term is -1.27 against -2.06, better by 0.79, and its volume and normalizer terms are -8.68 against -8.44, worse by 0.24. The Occam penalty for the second fault variable is real but small. The fit improves because two moderate biases, $+4.9$ and -5.1 metres, sit comfortably inside their five-degree priors where one bias of -8.5 does not, and because they let the source return to 114 kg/s from 122. The biases are poorly determined one by one, ± 3.9 and ± 2.9 metres, and correlated at 0.94: the readings fix their difference to ± 1.6 metres and leave their sum to the priors. More faults are not always penalized; two small faults can be more probable than one large one, and the evidence says when.

Exercise 12.4. With the transfer main's resistance freed around $16{,}049 \pm 3{,}210$, every single flow meter leaves its ratio ρ at 1.0000: no flow meter says anything about it. The expected information gains of the meters about the whole state still differ, from 1.79 nats for the meter on link 5–6 to 2.85 for the meter on the transfer main itself, and none of that information is about the resistance. The reason is that the transfer main is a bridge. The flow on a bridge is the total demand beyond it, whatever the bridge's roughness, and the flows elsewhere are set by the demands and by the resistances of the loops they lie in, which do not include the transfer main. Its resistance moves only the head drop across it, and flow meters do not read heads. A differential gauge across the main, reading 12.537 m to 0.01 m, does. Alone it brings ρ to 0.837, since the flow it would need to compare with is itself uncertain through the demands; together with the transfer-main flow meter it brings ρ to 0.088, since the pair reads the pipe's own head-loss law, flow against head drop. The meter worth most to the state is worth nothing to this parameter, which is Principle 12.3 in another form.

Chapter 13

Interventions and counterfactuals

An operator does not only ask what is. An operator asks what would happen: if that pipe were shut, if this demand were curtailed, if the setpoint were raised. Those are questions about a world that has been acted on, and a factor graph, which encodes compatibility among variables, does not by itself say what acting on the world does to it. This chapter supplies what is missing. The physics rows of Chapter 1 are mechanisms; the inputs of Chapter 4 are the quantities nothing in the model determines; and an intervention is either the imposition of an input or the replacement of a named mechanism by another. With that semantics, seeing and doing become different operations on the same graph, a counterfactual becomes three steps of which two are already familiar, and a decision becomes a tree of branches that share one posterior up to the point where they diverge. Each is worked on the book's models, and the difference between seeing and doing comes out in numbers.

13.1 Why the graph is not enough

Chapter 2 chose the factor graph over the directed graph because a closure has no direction: $Q = UA\,(T_h - T_c)$ relates three quantities and computes none of them. That choice served every chapter since. It has a price, and the price falls due here.

Conditioning on a reading is well defined on any factor graph: multiply by the likelihood, renormalize. It answers "given that I have seen $h_2 = 72$, what do I believe?" It does not answer "if I make $h_2 = 72$, what will happen?" The two differ because seeing $h_2 = 72$ is evidence about everything that could have caused it, the source and the room among them, while making $h_2 = 72$ causes it by some means and leaves the source and the trunk main as they were. To say what making does, one must say what mechanism was used to make it, and that is a statement about which relation in the model was overridden. The undirected graph does not contain that statement. Pearl's directed models contain it as the arrows: intervening on a variable severs the arrows into it and leaves the rest. The book's models have no arrows to sever, and something has to stand in for them.

13.2 Rows are mechanisms; inputs are exogenous

What stands in for them is the distinction Chapter 4 already drew. The variables of a model split into *inputs* $\mathbf{u}$, the quantities that carry priors because nothing in the model determines them (parameters, boundary conditions, sources, discrete availabilities), and *solved variables* $\mathbf{x}$,

which the physics determines from the inputs. Each physics row is a *mechanism*: a relation the system enforces, and one that could be enforced differently, by a different device, a different law, a controller. The well-posedness of §1.5 says that the rows, taken together, determine the solved variables from the inputs; no single row determines any single variable, but the set determines the set.

Definition 13.1 (Intervention)**.** An intervention on a physical model is one of two operations:

1. *Impose an input*: replace the prior on an input u_j by the fixed value $u_j = v$. The physics rows are untouched.
2. *Replace a mechanism*: remove a named physics row r_k and add in its place a new row r'_k that the intervening device enforces. The inputs and every other row are untouched.

The posterior after an intervention is the posterior of the modified model, with the evidence that was conditioned on before the intervention retained only where the intervention has not made it inapplicable.

The first operation needs no further choice; an input is exogenous by construction, and fixing it is what "do" means. The second needs a choice, and the definition makes the modeler state it: which row is replaced, and by what. "Make $h_2 = 72$" is not an intervention until it says whether the junction is held at 72 by a cooler that removes whatever heat is needed, which replaces the the valve closure, or by a room that is adjusted until the junction reads 72, which replaces the trunk main's prior. These are different devices, they leave different things unchanged, and they lead to different posteriors. A directed graph would have committed to one of them when its arrows were drawn. The factor graph defers the commitment to the moment of intervention, which is where it belongs.

> **Principle 13.1.** *Seeing is conditioning. Doing is replacing: an input's prior by a value, or a named mechanism by another. A value alone does not specify an intervention; the mechanism that imposes it does, and the modeler must name it.*

This is a structural-causal-model layer laid over the acausal graph. In Pearl's terms the physics rows are the structural equations, the inputs are the exogenous variables, and the replacement of a row is the surgery that "do" performs; what is different is that the equations are not written with an output variable, so the surgery must name the equation rather than the variable. That is not a limitation. It is the honest form of the operation for a physical system, in which the same variable can be set by different means.

13.3 Seeing versus doing, on the chain

Take the pumping chain with hard physics and the priors of Chapter 3. Every number is from the script `ch13_interventions.py` in the book's `scripts` directory. Table 13.1 sets four posteriors side by side.

Seeing. A gauge on the junction reads 72. The source's belief tightens from ±20 to ±5.8 and moves up; the trunk main's moves a little; the two become correlated; h_1 is predicted to ±1.3. The reading is evidence about its causes, and the posterior distributes it among them in proportion to how much each could have contributed. This is Chapter 3's result.

Table 13.1: The chain: prior predictive; conditioning on a reading of h_2; two interventions that both make $h_2 = 72$ by different mechanisms; and doing versus seeing on an input.

	m_{2a}	h_1	h_2	G	h_a
prior predictive	100 ± 20	90.0 ± 14.3	70.0 ± 10.4	100 ± 20	20.0 ± 3.0
see $h_2 = 72$ (sensor, acc. 0.5)	103.7 ± 5.8	92.7 ± 1.3	72.0 ± 0.5	103.7 ± 5.8	20.2 ± 2.9
do $h_2 = 72$, surface cooler replaces c_{2a}	100 ± 20	92.0 ± 4.0	72.0	100 ± 20	20.0 ± 3.0
do $h_2 = 72$, room control replaces π_{h_a}	100 ± 20	92.0 ± 4.0	72.0	100 ± 20	22.0 ± 10.0
do $G = 120$		104.0 ± 3.0	80.0 ± 3.0	120	20.0 ± 3.0
see $G = 120$ (meter, acc. 2)		103.9 ± 3.3	79.9 ± 3.2	120.0 ± 2.0	20.0 ± 3.0

Doing, one way. A cooler holds the junction at 72, removing whatever heat it must. The mechanism replaced is the the valve closure, which no longer describes how heat leaves the junction. The source is untouched: 100 ± 20, as under the prior. So is the room. The flow through the junction is still the source, by the balances, and the discharge head is 72 plus the source over the conductance: 92 ± 4, its spread now the source's spread through the conductance rather than the junction's. Nothing upstream of the junction has learned anything, because nothing was observed.

Doing, another way. The trunk main is regulated until the junction reads 72. The mechanism replaced is the trunk main's prior: the trunk main is no longer a forecast but a control variable set to whatever the surface needs, $h_a = 72 - G/k_2$, which is 22 ± 10 with the source's uncertainty passed through. The discharge head and the source are exactly as under the first intervention, since the junction is at 72 and the source is untouched either way; the trunk main is entirely different, and so is the physical device. Both are "do $h_2 = 72$". Neither is "see $h_2 = 72$".

Doing on an input. Set the source to 120 kg/s. There is no mechanism to name: the source is an input and setting it replaces its prior. The heads follow: $h_2 = 80 \pm 3$, the trunk main's spread passed through. Now instead *observe* the source at 120 with a meter accurate to two kg/s, and condition. The posteriors are the same to within the meter's accuracy. For an exogenous input, seeing and doing coincide, because an input has no causes in the model for a reading of it to be evidence about. That is the sense in which the inputs are exogenous, and it is why the first operation of Definition 13.1 needs no choice.

Proposition 13.1 (Seeing and doing on an input)**.** *Let u_j be an input whose prior is a factor on u_j alone. Whatever other evidence the model holds, the posterior over every other variable after a noiseless reading $u_j = v$ equals the posterior after imposing $u_j = v$.*

Proof. Conditioning on the reading multiplies the joint density by $\delta(u_j - v)$ and renormalizes. Under the delta, the prior factor $p_j(u_j)$ becomes the constant $p_j(v)$, which cancels in the normalization. Imposing the value replaces $p_j(u_j)$ by $\delta(u_j - v)$ directly. The two unnormalized products differ by the constant factor $p_j(v)$, so their normalized versions are the same. □

For a solved variable the argument fails, because conditioning keeps every row and doing removes one. The removed row carried information about the other variables, and the chain's two interventions show how much: each removes a different row and each leaves a different posterior. Figure 13.1 draws the three in the plane of the two inputs, which are what the worlds share. Seeing the junction at 72 confines the inputs to a thin ridge, $G/2 + h_a \approx 72$, correlated at -0.985:

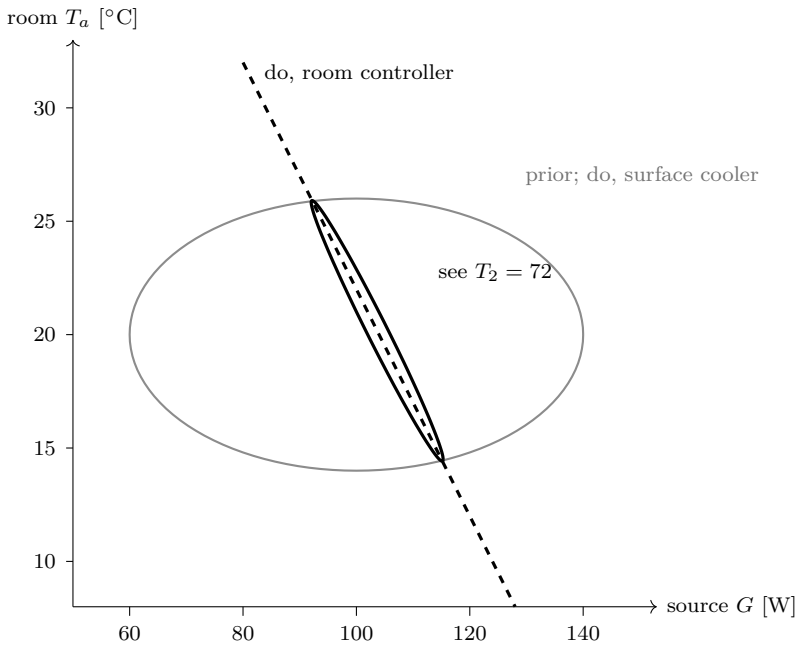


Figure 13.1: Seeing and doing on the chain, in the plane of the two inputs, at two standard deviations. Grey: the prior, which is also the input posterior after a surface cooler holds $h_2 = 72$. Black: after seeing $h_2 = 72$, a thin ridge. Dashed: after a room controller holds $h_2 = 72$, the line $h_a = 72 - G/2$, along which the source keeps its prior.

the reading is evidence, and it moves the belief about the causes. Doing it with a surface cooler leaves the inputs at their prior, since nothing was observed. Doing it with a room controller puts them on the line $h_a = 72 - G/2$ exactly, since the trunk main is now whatever the source requires. The source keeps its prior spread along the line, which is why the trunk main's spread becomes ± 10.

13.4 Counterfactuals: three steps

A counterfactual asks what would have happened, given what did. It combines evidence about the world as it is with an intervention on the world as it might be, and the combination has a definite order.

1. *Abduction.* Condition on the evidence in the factual model and read off the posterior of the inputs. This is what the evidence says about the exogenous quantities, which are what the two worlds share.
2. *Action.* Apply the intervention: impose the input or replace the mechanism.
3. *Prediction.* Push the abduced posterior of the inputs through the intervened model, with the evidence removed, since it was evidence about the factual world.

The evidence is used once, in step one, to learn about the inputs. It is not used in step three, because in the counterfactual world the readings would have been different; what carries over is not the readings but what they revealed about the demands, the parameters, the sources.

Proposition 13.2 (Counterfactuals in a linear model)**.** *Let the physics rows be linear and hard, let*

Table 13.2: The triangle: factual posterior with the meter reading; the counterfactual outage by abduction, action and prediction; and the naive re-conditioning of the outage model on the same reading.

	m_{12}	m_{23}	q_2	q_3
factual: pipe open, $y_{12} = 1.25$	1.25 ± 0.02	-0.35 ± 0.09	-1.60 ± 0.09	-0.55 ± 0.18
counterfactual: pipe shut, demands abduced	1.60 ± 0.09	0	-1.60 ± 0.09	-0.55 ± 0.18
naive: pipe shut, re-conditioned on y_{12}	1.25 ± 0.02	0	-1.25 ± 0.02	-0.50 ± 0.20

the factual posterior of the inputs be $\mathcal{N}(\mathbf{m}_u, \mathbf{\Sigma}_u)$, and let the intervened rows determine the solved variables as $\mathbf{x}' = \mathbf{A}'\mathbf{u} + \mathbf{b}'$. Then the counterfactual distribution of $\mathbf{x}'$ is $\mathcal{N}(\mathbf{A}'\mathbf{m}_u + \mathbf{b}',\ \mathbf{A}'\mathbf{\Sigma}_u\mathbf{A}'^{\mathsf{T}})$. The naive computation conditions the prior of the inputs on the readings through the intervened map in place of the factual one, and it agrees with the counterfactual only when the intervention leaves the readings' dependence on the inputs unchanged.

Proof. Abduction conditions the factual model on the readings. The inputs' posterior is Gaussian, since the factual model is linear-Gaussian, and it is $\mathcal{N}(\mathbf{m}_u, \mathbf{\Sigma}_u)$. After the action the solved variables are the linear function $\mathbf{A}'\mathbf{u} + \mathbf{b}'$ of the inputs, by the well-posedness of the intervened rows, and a linear image of a Gaussian is Gaussian with the mean and covariance stated. The naive computation treats a reading $y = \mathbf{h}^{\mathsf{T}}\mathbf{x}$ as $\mathbf{h}^{\mathsf{T}}(\mathbf{A}'\mathbf{u} + \mathbf{b}')$ where the factual world had $\mathbf{h}^{\mathsf{T}}(\mathbf{A}\mathbf{u} + \mathbf{b})$. The two conditionings coincide exactly when $\mathbf{h}^{\mathsf{T}}\mathbf{A}' = \mathbf{h}^{\mathsf{T}}\mathbf{A}$ and $\mathbf{h}^{\mathsf{T}}\mathbf{b}' = \mathbf{h}^{\mathsf{T}}\mathbf{b}$. □

The loop with a pipe shut. Pipe 1–2 is metered at 1.25 with the loop intact and the demands uncertain as in Chapter 7. What would the flows have been if pipe 2–3 had been out? Table 13.2 gives the three answers one could compute.

Abduction: the reading, with the loop intact, says the demand at junction 2 is -1.60 ± 0.09 and the demand at junction 3 -0.55 ± 0.18, correlated at -0.94 since the meter sees a combination of them. Action: replace the closure of pipe 2–3 by $m_{23} = 0$. Prediction: with the loop open, junction 2 is served by pipe 1–2 alone, so the metered pipe would carry the whole of junction 2's demand, 1.60 ± 0.09, a third more than it does, with the demand's uncertainty passed through. That is the counterfactual, and it is what an operator wants to know before sending a crew to shut the valve.

The naive computation conditions the outage model on the reading, as if the meter had read 1.25 with the pipe already shut. It concludes that the metered pipe would carry 1.25, which is what it does now, and that junction 2's demand is -1.25, which contradicts what the factual world established. It has used the reading as evidence in a world where the reading would not have occurred. The difference between the two rows is the whole content of the word "counterfactual", and it is a third of a per-unit on the pipe in question. In the terms of Proposition 13.2, the outage makes junction 2 branched, so the row of $\mathbf{A}'$ for the metered flow picks out the demand at junction 2 alone, and the counterfactual flow is that demand's abduced posterior with its sign changed. The meter's row under the outage reads $-q_2$ where the factual one read a mixture of both loads, so the naive computation is conditioning on a different functional of the inputs. Figure 13.2 draws the counterfactual flow against the limit: 49 percent of it lies beyond. The naive answer, a spike at the factual flow, would have told the operator the pipe was safe.

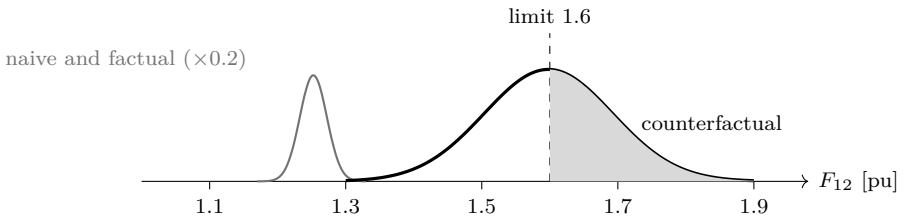


Figure 13.2: The flow on pipe 1–2 had pipe 2–3 been out, given the factual reading of 1.25 with it in. Black: the counterfactual, 1.60 ± 0.09, with the 49 percent beyond the limit shaded. Grey, drawn at a fifth of its height: the naive re-conditioning, which coincides with the factual flow.

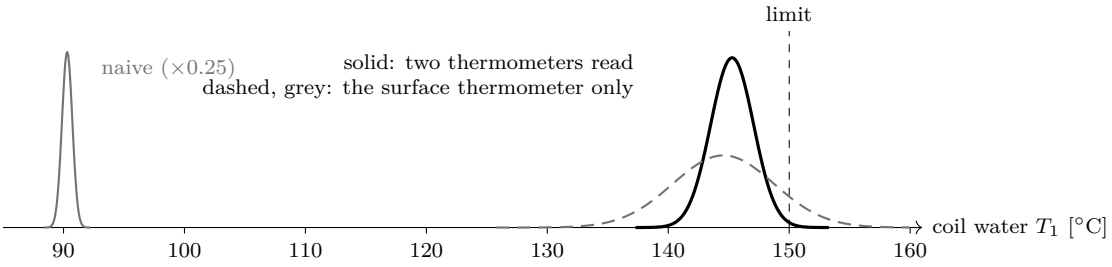


Figure 13.3: The discharge head had the fan run at half speed. Solid: the counterfactual with both gauges in the factual world, 145.3 ± 1.8. Dashed: with the junction gauge only, 144.6 ± 4.2, of which 9.7 percent lies beyond the 150°C limit. Grey, drawn at a quarter of its height: the naive re-conditioning at 90.3, which bends the inputs to fit the old readings.

> **Principle 13.2.** *A counterfactual learns the inputs from the factual evidence, changes the mechanism, and predicts with the evidence removed. Re-conditioning the changed model on the old evidence answers a different question, and usually the wrong one.*

The chain with the valve half shut. The same distinction on the pumping chain, where the mechanism replaced is a closure rather than a prior. Both gauges of the chain read, 72 and 93 m, with the valve's conductance at 2 kg/s per metre. What would the heads have been had the fan run at half speed, the conductance at 1? Abduction gives the inputs the chain's posterior: the source at 104.63 ± 2.86 and the trunk main at 19.73 ± 1.73, correlated at -0.979. Action replaces the the valve closure by $m_{2a} = 1.0\,(h_2 - h_a)$. Prediction pushes the abduced inputs through: the junction would have been at 124.4 ± 1.2 and the discharge at 145.3 ± 1.8, with a probability of 0.004 of exceeding 150°C. The naive computation re-conditions the half-speed model on the two readings and concludes that the junction would have read 75.3, the discharge 90.3, the source 75 kg/s and the trunk main 0°C. It has bent the inputs until the old readings fit the new fan, which answers the question of what inputs would have produced these readings at half speed, and nobody asked it. With only the junction gauge in the factual world the counterfactual body head is 144.6 ± 4.2 and the probability of exceeding 150°C is 0.097. The second gauge, by pinning the ridge, cuts that probability by a factor of about twenty-four: a counterfactual is only as sharp as its abduction. Figure 13.3 draws the three answers.

Table 13.3: Four actions on the triangle, evaluated on the shared abduced posterior of the demands. Limit on pipe 1–2: 1.6 kg/s.

action	m_{12}	$\mathbb{P}(m_{12} > 1.6)$	m_{13}
do nothing	1.25 ± 0.02	0.000	0.90
shut pipe 2–3	1.60 ± 0.09	0.49	0.55
curtail 0.3 at junction 2	1.05 ± 0.02	0.000	0.80
shut 2–3 and curtail 0.3	1.30 ± 0.09	0.001	0.55

13.5 Decisions as branches

An operator choosing among actions asks the counterfactual question once per action, and the three steps have a structure that makes this cheap. Abduction is done once: the posterior of the inputs, which every action shares, is the one computation on the factual evidence. Each action is then a branch: an intervened model, evaluated with the shared input posterior. The branches form a tree whose root is the factual posterior, whose edges are interventions, and whose leaves are the worlds that result; a second intervention under a first is a child branch, and a contingency that may or may not occur under an action is a pair of children weighted by its probability. The readouts of Chapter 12 apply at every leaf, and a weighted reduction over leaves, a mean, a tail quantile, an expected shortfall, gives the action's risk.

The do-nothing baseline. One branch is always the identity: no intervention, the factual posterior carried forward. It is the baseline against which every action is judged, and its inclusion is not a formality. An action is worth taking only if its leaf is better than the baseline's, and "better" is a reduction over the leaf's posterior that must be computed for the baseline too, with the same inputs and the same evidence. Recommending an action without the do-nothing leaf beside it is recommending it against nothing.

Four branches on the triangle. The quantity at risk is the flow on pipe 1–2 against a limit of 1.6 kg/s, the most that main may carry before its velocity starts to scour the lining. Four actions: nothing; shut pipe 2–3; curtail junction 2 by 0.3; both. Table 13.3 gives each leaf's flow and the probability of exceeding the limit, all from the one abduced posterior of the demands.

Shutting the link alone puts the metered pipe at its limit with even odds of exceeding it; the loop was carrying a third of junction 2's demand around the other way. Curtailment alone is safe and unnecessary. Opening the link with curtailment is safe, at a tenth of a percent, and it is the branch an operator who needs the link shut for a repair would choose. The four leaves cost four solves on a shared prior; on a large network they cost four region messages, with the abduction and every region the actions do not touch reused across the tree, as Chapter 8 arranged.

Figure 13.4 draws the tree, with a contingency added under both actions that shut pipe 2–3 (Exercise 13.4): pipe 1–3 fails with probability 0.02. With both pipes shut, junction 3 is cut off and its demand, 0.55 ± 0.18 kg/s, is unserved. The flow on pipe 1–2 is unchanged, since junction 2 was already fed by that pipe alone. The contingency leaves the probability of an overload where it was and adds an expected unserved demand of 0.011 kg/s to both branches. A leaf must report every quantity that matters, not only the one that motivated the tree.

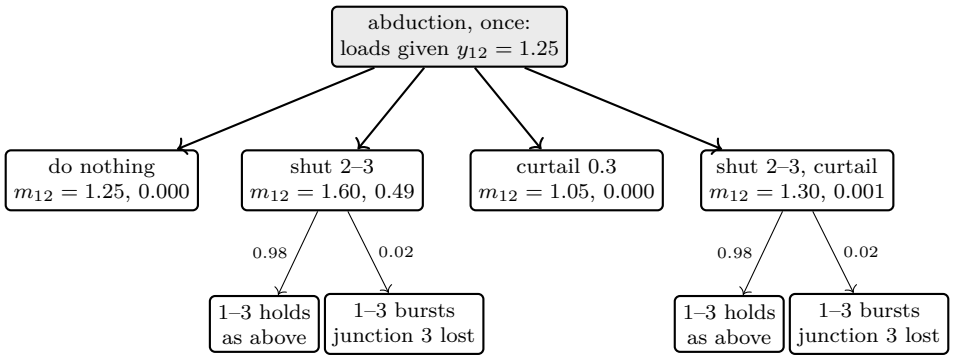


Figure 13.4: The decision tree of Table 13.3. The root is the abduction, done once; each edge below it is an intervention; each leaf is evaluated on the shared input posterior and shows the flow on pipe 1–2 and the probability that it exceeds its limit of 1.6. Under the two actions that shut pipe 2–3, a contingency splits the leaf: pipe 1–3 fails with probability 0.02, islanding junction 3.

Table 13.4: Five branches on the two treatment works in a drought, on 200,000 shared draws from the abduced inputs: the expected unserved demand, the probabilities of any shortfall and of one above 50 litres a second, the change against doing nothing, and that change's standard error computed on the shared draws and on independent ones.

action	$\mathbb{E}[U]$ [L/s]	$\mathbb{P}(U > 0)$	$\mathbb{P}(U > 50)$	change [L/s]	s.e., shared	s.e., independent
do nothing	30.45	0.554	0.263	—	—	—
standby stream	2.30	0.085	0.013	−28.15	0.086	0.093
transfer main +10%	30.14	0.545	0.261	−0.31	0.007	0.128
restrict 100 L/s	2.27	0.081	0.014	−28.17	0.078	0.094
standby stream and main	2.30	0.085	0.013	−28.15	0.086	0.093

13.6 Worked example: a drought week

The two treatment works of Chapter 10 face a drought. The forecast puts the river drawdown at 8.5 ± 0.7 centimetres, and that is the abduction here: the one piece of evidence, folded into the inputs' posterior, while the stream outputs and main capacities keep their priors. The city's demand is 2,100 litres a second. Besides doing nothing, the control room has four actions: start works 1's standby treatment stream, a fourth stream rated like the others; raise works 2's transfer main capacity by 10 percent, which a booster set can do; cut 100 litres a second of demand by restriction; or start the standby stream and raise the main. Each is a branch, and each is evaluated on the same 200,000 draws from the abduced inputs, which is the Monte Carlo form of a shared posterior. Table 13.4 gives the leaves.

Doing nothing leaves an even chance of a shortfall and a one-in-four chance of more than 50 litres a second. The standby stream cuts the expected shortfall from 30.5 to 2.3 litres a second and the chance of any to 8.5 percent, as much as restricting 100 litres a second of demand does, without restricting anything. Raising the transfer main does almost nothing, 0.31 litres a second, and adding it to the standby stream does nothing at all. The reason is in the do-nothing leaf: at this drawdown both works are limited by their treatment streams in 93 percent of the draws, and a wider main downstream of a treatment-limited works carries no more water. The model has answered a question the control room did not ask, which constraint binds, and that answer

decides between two actions that sound equally plausible.

The last two columns are about the arithmetic of branches. The decision rests on the differences between leaves, and computing every leaf on the same draws makes the noise of a difference small where the actions are similar. The main's effect, -0.31 litres a second, has a standard error of 0.007 on the shared draws and 0.128 on independent ones: the independent route would need 388 times the draws to resolve it as well. For the standby stream, whose leaf is far from the baseline, sharing the draws helps little, since the difference is almost all of the baseline's own variability. This is the sampling counterpart of sharing the abduction: the branches share not only the posterior but the samples of it, and the comparison is sharper for it.

Proposition 13.3 (Shared draws)**.** *Let two branches be evaluated on N draws each, giving estimates $\bar{U}_a$ and $\bar{U}_0$ of their means, and let σ_a and σ_0 be the spreads of their outcomes. With independent draws,* $\mathrm{Var}(\bar{U}_a - \bar{U}_0) = (\sigma_a^2 + \sigma_0^2)/N$*. With shared draws it is* $(\sigma_a^2 + \sigma_0^2 - 2\rho\,\sigma_a\sigma_0)/N$*, where ρ is the correlation of the two outcomes across draws. Sharing helps exactly when $\rho > 0$.*

Proof. With shared draws the difference of the means is the mean of the N differences $U_a(\mathbf{x}_n) - U_0(\mathbf{x}_n)$, whose variance is $\mathrm{Var}(U_a - U_0)/N = (\sigma_a^2 + \sigma_0^2 - 2\,\mathrm{Cov}(U_a, U_0))/N$. With independent draws the two means are independent and their variances add. □

For the transfer main, the two leaves differ by a third of a litre a second on draws whose shortfall varies by tens of litres a second, so ρ is nearly one and sharing is worth a factor of 388. For the standby stream the leaves are nearly uncorrelated, since the stream removes most of the shortfall wherever it occurs, and the factor is close to one.

Figure 13.5 splits the leaves by the drawdown, a partial sum on the shared draws. The main's curve lies on the do-nothing curve at every drawdown. The standby stream and the restriction are close overall, but they are not the same action. At a milder drawdown a shortfall happens when some works' streams are weak, and works 1's transfer main caps what a standby stream there can add; restricting demand helps wherever the shortfall comes from, and it does slightly better. At the deepest drawdowns both works are derated together, the standby stream adds as much as works 1's main allows, about 130 litres a second at the mean capacity, more than the 100 restricted, and it does better.

13.7 Worked example: holding the transfer flow

An operator wants the transfer main of the two-district network held at 19.0 kg/s, below the 21.0 the demand priors imply. Definition 13.1 requires a device, and two are candidates. A booster pump on the transfer main adds an adjustable lift ϕ to that pipe's closure, $h_3 - h_4 + \phi = K_{34}Q_{34}(Q_{34}^2 + q_0^2)^{0.426}$, and its controller sets ϕ to hold the flow. A local source at junction 4 — a borehole, or a service reservoir drawn down on demand — adds a source g_4 to that junction's balance, and its controller sets g_4 to hold the flow. Each intervention adds the controller's target, $m_{34} = 19.0$, and frees the control variable it acts through. Nothing is observed: there are no meters, only the five demand priors. Every number is from `ch13_transfer.py`.

The local source does what was asked. The demands are untouched: the sum of district B's three demands is 21.000 ± 3.464 kg/s after the intervention, exactly what it was before and exactly what the three priors say on their own. The source's duty is 2.000 ± 3.464 kg/s; it absorbs whatever the demands do, and its spread is theirs because it has no independent belief of its own. The booster pump does something else. With nothing observed, the posterior of district

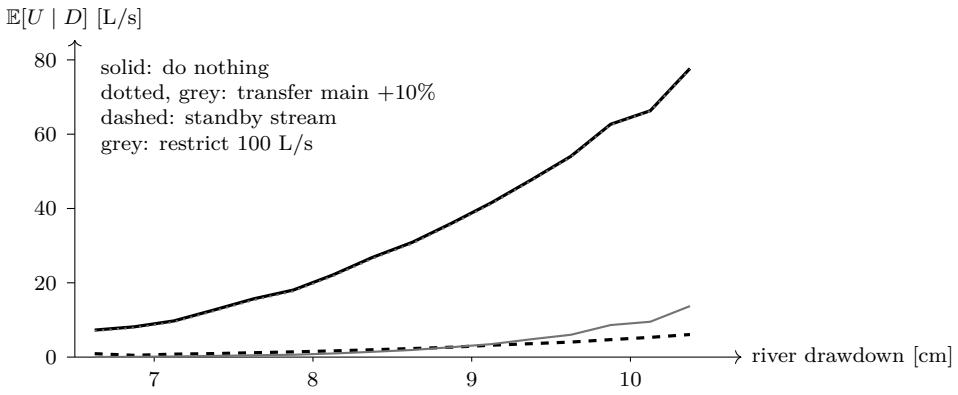


Figure 13.5: The expected unserved demand given the river drawdown, for four branches of Table 13.4, read from the shared draws in bins of a quarter centimetre. The main's curve lies on the do-nothing curve. The standby stream and the restriction cross: restricting does slightly better at milder drawdowns and the standby stream at the deepest.

B's demands moves: their sum goes from 21.000 ± 3.464 to 19.000 ± 0.017, and each demand falls by 0.667 kg/s with its spread dropping from 2.000 to 1.633. The pump's lift, meanwhile, is left wherever its prior put it, ± 1000 m: the model has no use for it. The demands are exogenous, and an intervention has conditioned them.

The reason is the bridge. The flow on a bridge is the total demand beyond it — that is mass conservation, not an approximation of anything — so no amount of pump lift can change it. Raise the head at junction 4 and the district's pressures rise; the water it draws does not. The model says so twice, by leaving ϕ undetermined and by bending the demands to meet the target. Figure 13.6 draws district B's demand sum under the two devices.

Proposition 13.4 (Realizable interventions)**.** *Let the physics rows be linear. Let an intervention replace mechanism rows, possibly adding control variables. If the modified rows determine the solved and control variables uniquely for every value of the inputs, then, with no evidence, the posterior of the inputs after the intervention is their prior. If they determine them only for inputs on a proper subspace, the posterior of the inputs is confined to that subspace.*

Proof. Write the modified rows as $\mathbf{G}_x\mathbf{x} + \mathbf{G}_u\mathbf{u} = \mathbf{c}$, with $\mathbf{x}$ now including the control variables. If $\mathbf{G}_x$ is square and nonsingular, then $\mathbf{x} = \mathbf{G}_x^{-1}(\mathbf{c} - \mathbf{G}_u\mathbf{u})$ for every $\mathbf{u}$, and the joint density is the prior of $\mathbf{u}$ times a delta on $\mathbf{x}$ whose integral over $\mathbf{x}$ is the constant $1/|\det \mathbf{G}_x|$. The marginal of $\mathbf{u}$ is its prior. If $\mathbf{G}_x$ has a left null vector $\boldsymbol{\ell}$, then $\boldsymbol{\ell}^\mathsf{T}(\mathbf{c} - \mathbf{G}_u\mathbf{u}) = 0$ is a condition on the inputs alone, which every solution requires, and the marginal of $\mathbf{u}$ is the prior restricted to that hyperplane. With soft rows the restriction is a Gaussian of the rows' width about the hyperplane. □

The proposition asks for linear rows and the closures here are not, but the balances are, and the balances are the whole of the argument. For the booster pump, the left null vector sums the balances of district B and the target row: the balances say $m_{34} = d_4 + d_5 + d_6$, the target says $m_{34} = 19.0$, and together they say $d_4 + d_5 + d_6 = 19.0$, a condition on inputs alone. Not one closure appears in it, which is why adding a pump to a closure cannot escape it. That is the hyperplane to which the demands were confined, at the width of the balance rows: the posterior spread of 0.017 kg/s is the three balance widths of 0.01 in quadrature, and each demand's spread of 1.633 is $2\sqrt{2/3}$, the prior spread projected onto the plane.

Principle 13.3. *An intervention that moves the posterior of an exogenous input, with nothing observed, is not realizable by the named device. Check it before reading any other number: after an intervention, and before any evidence, the inputs must be where their priors put them.*

The local source's intervention sits inside district B: district A's message to the transfer main is unchanged and is reused, and only district B's is recomputed, with g_4 in it. That is the reuse of Chapter 8 applied to decisions: an action changes the messages of the regions it touches and no others.

13.8 What the semantics does not decide

Three things are outside what Definition 13.1 settles, and the reader should know where the boundary is.

It does not decide which row a real device replaces. A cooler on the surface cooler replaces the the valve closure in the example because the example said so; a real valve may also change the junction's stored volume, add a fan whose power appears as a source, and fail. The definition requires the modeler to state the replacement; it does not validate it.

It does not decide what evidence survives an intervention. The rule given, that evidence is retained where the intervention has not made it inapplicable, is a rule about which readings still describe the counterfactual world, and applying it requires judgment: a meter on a pipe that is shut reads nothing afterward; a meter on a pipe elsewhere reads something, but not what it read before. The three-step recipe sidesteps the question by using the evidence only in abduction, which is the safe default, and the modeler who wants to retain a reading in the counterfactual world should say why it would still hold.

And it does not, by itself, quantify how well the model's mechanisms correspond to the world's. An intervention computed on a wrong model is a wrong prediction with an honest error bar; the diagnostics of Chapter 12 say whether the factual model fits, and nothing in this chapter says whether the counterfactual one would.

13.9 In practice

Name the device, then the row it replaces. "Set h_2 to 72" is a wish. "A slab cooler replaces the the valve closure" is an intervention. Write the second form down for every action, including the control variable the device acts through.

Check that the inputs did not move. Before reading any other number, compare the inputs' posterior after the intervention, with no evidence, against their prior. If it moved, the device cannot do what the row claims, as the booster pump on a bridge could not.

Abduce once, branch many, and keep the do-nothing leaf. The abduction is the only step that sees the evidence, and every branch shares it. The do-nothing branch is the baseline every recommendation is judged against.

Drop the evidence in prediction. A reading describes the factual world. Carry it into a counterfactual only when you can say why the device leaves it unchanged, and say so in the report.

Evaluate every branch on the same draws. When the leaves are computed by sampling, share the samples across branches, as the branches share the abduction. For two actions whose leaves are close, this is worth hundreds of times the draws; for actions far apart it costs nothing.

Report every quantity at every leaf. A contingency may leave the quantity that motivated the tree unchanged and move another, as pipe 1–3's failure left the overload probability alone and shed the demand at junction 3.

Sharpen the abduction to sharpen the counterfactual. A counterfactual inherits the spread of the inputs' posterior. The second gauge on the chain cut the probability of the half-speed body head exceeding its limit from 0.097 to 0.004.

13.10 What is missing

Every branch here reported a mean, a spread and a probability. What it did not report is why: which uncertain input the risk depends on most, how much a better forecast of the demand at junction 3 would change the verdict, and whether the probability of 0.49 is a floor, a ceiling or a point estimate. Those are questions of sensitivity and attribution, and they need the solve to return its gradient. That is Chapter 14. And every intervention here was one pipe or one demand; Chapter 15 asks about the failures the operator did not choose, many at once.

Notes and further reading

The distinction between seeing and doing, the "do" operator, the surgery on structural equations that defines it, and the three-step abduction-action-prediction account of counterfactuals are Pearl's [70]; Peters, Janzing and Schölkopf [72] give a compact modern treatment with structural causal models as the primary object. In those accounts each structural equation has a designated output and the intervention severs the arrows into it. The version in Definition 13.1, with the equation named rather than the variable, is the book's adaptation to models whose equations are acausal, and it is what a physical intervention amounts to: a device that enforces a different relation. The reading of a Chapter 1 model as a structural causal model whose exogenous variables are the inputs and whose mechanisms are the rows is the book's, and it is the reason the input/solved-variable split of Chapter 4 was drawn where it was.

Decision trees over interventions and contingencies, evaluated by weighted reductions over leaves, are standard in reliability and risk analysis; expected shortfall as the tail reduction is Rockafellar and Uryasev's [77], and Part VI uses it. The do-nothing baseline as a mandatory branch is a rule of this book's case-study practice, adopted after enough reports had recommended actions against no comparison.

Summary

- A factor graph encodes compatibility, not mechanism; it cannot say what acting does until the mechanism is named.
- Physics rows are mechanisms, inputs are exogenous. An intervention imposes an input or replaces a named row. "Do $h_2 = 72$" by a surface cooler and by a room controller are

different interventions with different posteriors; seeing $h_2 = 72$ is a third. On an input, seeing and doing coincide.

- A counterfactual is abduction, action, prediction, with the evidence used only in the first. On the triangle the counterfactual outage flow is 1.60 ± 0.09; naive re-conditioning gives 1.25, the factual flow, and is wrong.
- Decisions are branches on a shared abduced posterior, with the do-nothing leaf always present. Closing the link alone: even odds of exceeding the limit; with curtailment: a tenth of a percent.
- In a linear model a counterfactual is the abduced input posterior pushed through the intervened map. Had the chain's fan run at half speed, the discharge would have been at 145.3 ± 1.8; the naive computation says 90.3 and bends the trunk main to 0°C.
- An intervention is realizable only if it leaves the exogenous inputs where their priors put them. A booster pump on a bridge cannot hold its flow, and the model shows it by moving the demands; a local source inside district B can.
- On the two treatment works in a drought the standby stream is worth as much as restricting 100 litres a second and the transfer main nothing, because both works are treatment-limited. Evaluating the branches on shared draws resolves the main's 0.3-litre effect with 388 times fewer samples.
- A leaf reports every quantity: a contingency under "close link 2–3" leaves the overload probability at 0.49 and adds an expected unserved demand of 0.011.
- The semantics requires the modeler to name the replaced mechanism and to decide what evidence survives; it does not do either for them.

Exercises

Exercise 13.1. On the chain, define a third intervention that makes $h_2 = 72$: a heater at node 2 that adds whatever power is needed, which adds a source term to the balance at node 2 and leaves both closures intact. Compute its posterior and compare h_1, m_{2a} and h_a with the two interventions of Table 13.1.

Exercise 13.2. Show that for an input with an independent prior, conditioning on a noiseless reading of it and imposing it give identical posteriors over every other variable, and that for a solved variable they do not in general. State the condition on the graph under which they coincide for a solved variable.

Exercise 13.3. Repeat the counterfactual of Table 13.2 with a second meter on pipe 1–3 in the factual world. Show that the abduced posterior of the loads is now tighter and less correlated, and that the counterfactual outage flow's spread falls accordingly. What does the naive computation give with two meters?

Exercise 13.4. Add to the branch tree of Table 13.3 a contingency: under the "shut pipe 2–3" action, pipe 1–3 also fails with probability 0.02. Compute the leaf for that contingency, the weighted probability of exceeding the limit on pipe 1–2 over the two leaves, and the expected shortfall of m_{12} at the 95 percent level.

Exercise 13.5. For the two-district network of Chapter 8, formulate the intervention "hold the transfer flow at 19.0 kg/s" in two ways: by a booster pump that modifies the transfer main's closure, and by curtailment in district B that shifts the demand priors. Compute both posteriors, identify which region's message each intervention changes, and say what is reused.

Solutions to selected exercises

Exercise 13.1. The heater adds its power s to the balance at node 2, $m_{2a} = m_{12} + s$, and its controller sets s so that $h_2 = 72$. The replaced mechanism is the free choice of s, and both closures stay. The source is untouched at 100 ± 20, and so is the trunk main at 20 ± 3. The flow closure gives $h_1 = 72 + G/5 = 92.0 \pm 4.0$, as in both interventions of Table 13.1. The the valve closure gives $m_{2a} = 2(72 - h_a) = 104 \pm 6$, whose spread is now the trunk main's rather than the source's, and the heater's power $s = m_{2a} - G = 4 \pm 21$ carries both. Against the junction cooler, h_1 is the same and m_{2a} differs, 104 ± 6 against 100 ± 20. Against the trunk main controller, the room differs, 20 ± 3 against 22 ± 10. Three devices, one value, three posteriors.

Exercise 13.2. The first part is Proposition 13.1. For a solved variable x_i, seeing keeps every row and adds $\delta(x_i - v)$, while doing by a device removes the row the device overrides and adds the same delta. The two coincide when the removed row, with the delta and every other factor in place, contributes a constant. That happens when the row contains a variable that appears in no other factor and has a flat prior: integrating the row over that variable gives the same number whatever the other variables are. A controller's actuator is such a variable. The heater of Exercise 13.1 is the example. Its power s appears only in the balance at node 2, with a flat prior, and holding $h_2 = 72$ with the heater gives exactly the posterior of seeing $h_2 = 72$ in the model where the heater's power is free, since the rows are the same. The junction cooler and the trunk main controller are not of this form in the original model. The the valve closure ties the trunk main to the source, and the trunk main's prior carries a forecast, so removing either changes what the other variables may be, and the posteriors of Table 13.1 differ from seeing.

Exercise 13.3. The second meter reads 0.90, where the one-meter posterior predicted 0.899 ± 0.092. With both meters the abduced loads are $q_2 = -1.597 \pm 0.047$ and $q_3 = -0.552 \pm 0.047$, correlated at -0.77 instead of -0.94: each meter reads a different combination of the demands, and together they pin both. The counterfactual outage flow on pipe 1–2 is $-q_2$, 1.597 ± 0.049, half the spread of the one-meter answer. Its probability of exceeding 1.6 is 0.47, barely moved, because the mean sits almost exactly at the limit. The naive computation conditions the outage model on both readings, which under the outage read $-q_2$ and $-q_3$ directly. It concludes $m_{12} = 1.25$, $q_2 = -1.25$ and $q_3 = -0.89$, now wrong for both loads. More factual evidence sharpens the counterfactual and leaves the naive answer as wrong as before.

Exercise 13.4. Under "shut pipe 2–3" the leaf without the contingency has $m_{12} = 1.60 \pm 0.09$ and a probability of 0.49 of exceeding 1.6. With pipe 1–3 also out, junction 3 is cut off: its demand of 0.55 ± 0.18 kg/s is unserved, and pipe 1–2 still carries junction 2's demand alone, the same 1.60 ± 0.09. The weighted probability over the two leaves is therefore still 0.49. For a Gaussian the mean of the worst five percent is $\mu + \sigma\,\varphi(z_{0.95})/0.05$, with φ the standard normal density and $z_{0.95} = 1.645$, so the expected shortfall of m_{12} at 95 percent is 1.79 kg/s, the same in both leaves. What the contingency adds is a unserved demand, expected $0.02 \times 0.55 = 0.011$ per unit. Under "shut 2–3 and curtail 0.3" the same contingency loses the same load, and the flow's expected shortfall is 1.49, below the limit.

Exercise 13.5. The booster pump is worked in §13.7. It modifies the transfer main's closure, so it changes the relation on the port itself, and with nothing observed it moves the district-B

demands to a sum of 19.000 ± 0.017 while leaving its own lift undetermined: by Proposition 13.4 it is not realizable, because the flow on a bridge does not depend on the head across it. Curtailment that shifts the demand priors, by 2.0 kg/s of expected district-B demand spread pro rata, changes only district B's factors, so only district B's message is recomputed and district A's is reused. It leaves the transfer flow at 19.000 ± 3.464 kg/s: the mean is where it was asked to be, but the flow is not held, since the demands still vary as much as they did. Holding it exactly takes a controller with an actuator inside district B, the local source at junction 4 of §13.7, which again changes only district B's message and absorbs the demands' variability in its own duty, 2.000 ± 3.464 kg/s.

Chapter 14

Attribution, sensitivity, and certificates

The previous chapter left three questions unanswered at every leaf of its decision tree: which uncertain input the risk depends on most, how much a unit change in each would move it, and whether a reported number is a floor, a ceiling or a guess. All three are questions about how a consequence of the physics varies with the inputs, and all three become cheap when the solve returns, along with the consequence, its gradient with respect to every input. This chapter shows where that gradient comes from, what attribution means and how it differs from sensitivity, and how a census of solves with gradients, harvested once, answers the three questions and several more at no further cost, with a certificate, whenever the consequence is convex in the inputs. It ends with the one trap that a certificate cannot survive: a gradient taken across a kink.

14.1 Three questions, one derivative

Let $G(\mathbf{u})$ be a scalar consequence of the physics, the shortfall, the transfer flow, the head of a component, as a function of the uncertain inputs $\mathbf{u}$. The operator's three questions are, in order of how much they ask of G:

1. *Sensitivity*: by how much does G change per unit change in u_i at the current state? That is $\partial G/\partial u_i$, one number per input, at one point.
2. *Attribution*: how much of the uncertainty in G is due to the uncertainty in u_i? That is a property of G over the whole prior, not at a point, and it needs the joint behavior of G and the inputs.
3. *Certification*: is the reported expectation, tail or worst case of G a bound, and in which direction? That needs a global property of G, and convexity is the one that delivers.

Sensitivity is the derivative. Attribution can be approximated by it and is exactly it only for a linear G. Certification uses it as a supporting plane. All three start from the same object, and the first task is to obtain it without differencing.

14.2 Sensitivities from the solve

At a solved state $\mathbf{F}(\mathbf{z}^\star, \mathbf{u}) = \mathbf{0}$ the solved variables are an implicit function of the inputs, $\mathbf{z} = X(\mathbf{u})$, and the implicit function theorem gives their derivative from the two Jacobians the solve already

assembled:

$$\frac{\partial X}{\partial \mathbf{u}} \;=\; -\,\mathbf{J}_z^{-1}\,\mathbf{J}_u, \qquad \mathbf{J}_z = \frac{\partial \mathbf{F}}{\partial \mathbf{z}}, \quad \mathbf{J}_u = \frac{\partial \mathbf{F}}{\partial \mathbf{u}}. \tag{14.1}$$

$\mathbf{J}_z$ is the Jacobian of Chapter 1, square and already factored; $\mathbf{J}_u$ is the same assembly with the input columns kept rather than dropped, which costs nothing extra. For n_u inputs the forward form is n_u solves against the held factorization, one per column of $\mathbf{J}_u$. For one consequence $G = \mathbf{c}^\mathsf{T}\mathbf{z}$ the *adjoint* form is a single solve, $\mathbf{J}_z^\mathsf{T}\boldsymbol{\lambda} = \mathbf{c}$, after which $\partial G/\partial \mathbf{u} = -\boldsymbol{\lambda}^\mathsf{T}\mathbf{J}_u$ for every input at once. A model with a hundred inputs and one consequence pays one solve; with one input and a hundred consequences, one solve the other way.

Derivation. Differentiate the identity $\mathbf{F}(X(\mathbf{u}), \mathbf{u}) = \mathbf{0}$ with respect to $\mathbf{u}$: $\mathbf{J}_z\,\partial X/\partial \mathbf{u} + \mathbf{J}_u = \mathbf{0}$. At a well-posed solution $\mathbf{J}_z$ is nonsingular, which gives equation (14.1). For $G = \mathbf{c}^\mathsf{T}\mathbf{z}$ the derivative is $\partial G/\partial \mathbf{u} = -\mathbf{c}^\mathsf{T}\mathbf{J}_z^{-1}\mathbf{J}_u$. Grouped from the left, $\boldsymbol{\lambda}^\mathsf{T} = \mathbf{c}^\mathsf{T}\mathbf{J}_z^{-1}$ is the solution of $\mathbf{J}_z^\mathsf{T}\boldsymbol{\lambda} = \mathbf{c}$, one solve, after which $-\boldsymbol{\lambda}^\mathsf{T}\mathbf{J}_u$ is a sparse product. Grouped from the right, $\mathbf{J}_z^{-1}\mathbf{J}_u$ is n_u solves, one per input column. □

On the pumping chain with the conductance k freed, the inputs are (G, h_a, k); here G is the chain's source, not the consequence $G(\mathbf{u})$ of §14.1. The script `ch14_attribution.py` in the book's `scripts` directory, which supplies every number in the chapter, evaluates equation (14.1) at the nominal state:

$$\frac{\partial (h_1, h_2)}{\partial (G, h_a, k)} \;=\; \begin{pmatrix} 0.700 & 1.000 & -4.000 \\ 0.500 & 1.000 & 0 \end{pmatrix}, \tag{14.2}$$

agreeing with central differences to 2×10^{-9}. The discharge head rises 0.7 metres per kg/s of source, one degree per degree of room, and falls four metres per unit of conductance; the surface does not see the conductance at all. The same numbers are the lever arms that Chapters 3 and 11 read off by hand, now read off the Jacobian for any model.

At scale. On the chain, with three inputs, the choice between the forward and adjoint forms hardly matters. Take instead the grid network of Chapter 11 at 100×100 nodes, with an uncertain source at every node: ten thousand unknowns, ten thousand inputs, and one consequence, the head at the centre, where a component sits. The forward form would solve once per source: ten thousand solves against the held factorization. The adjoint form solves once, and returns all ten thousand sensitivities. Ten thousand to one is the whole of the adjoint's claim, and unlike a wall clock it is the same on every machine; the script prints the clock too, for whoever wants it. At the nodes checked the sensitivities agree with finite differences to six digits. The centre warms by 0.123 metres per kg/s of its own source, 0.063 per kg/s of a neighbour's and 0.0025 per kg/s of a source ten nodes away. The sensitivities sum to 4.000 metres per kg/s, which is $1/h$: a uniform extra kg/s at every node must raise every node by $1/h$ for the grid to reject it. And the centre follows the circuit water head one for one. Figure 14.1 draws the sensitivities around the centre. The map is the grid's Green's function seen from the centre, and since the grid is linear it is also the attribution: with independent sources of equal spread, each source's share of the centre's variance is proportional to the square of its sensitivity, by Proposition 14.1.

14.3 Attribution

Attribution asks for a decomposition of $\mathrm{Var}[G]$ into shares due to each input. Two definitions are standard.

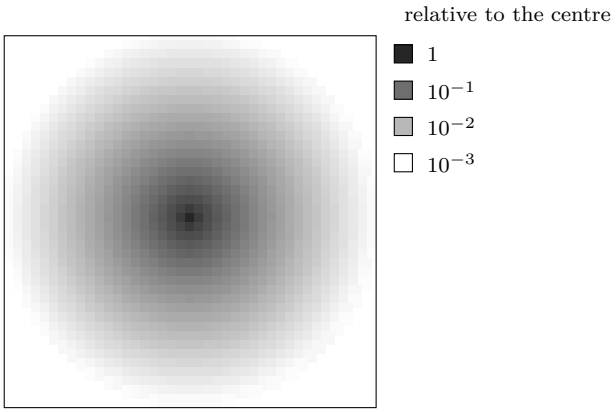


Figure 14.1: The sensitivity of the centre head of a 100×100 grid network to the water source at each node within twenty nodes of the centre, relative to the centre's own, on a logarithmic grey scale. One adjoint solve computes all ten thousand values.

The *Sobol* first-order index of input i is the fraction of the variance explained by the conditional expectation given u_i alone, $S_i = \mathrm{Var}[\mathbb{E}[G \mid u_i]]/\mathrm{Var}[G]$; the *total* index adds every interaction involving u_i. When G is additive in independent inputs the first-order indices sum to one and equal the totals; when it is not, the difference measures interaction, and the indices do not by themselves say how to divide the interaction among the inputs involved.

The *Shapley* share does. It averages, over every ordering of the inputs, the increase in explained variance that input i brings when added to the inputs before it. Shares sum to one, respect symmetry, and give an input that matters only jointly with another half the joint credit. For d inputs the average runs over $d!$ orderings and 2^d subsets, each subset needing a conditional expectation; that is the computation Chapter 10 found to cost the sampling route $(d+4)N$ solves and the grid route nothing beyond the grid.

The cheap proxy is the gradient, and for a linear consequence it is exact.

Proposition 14.1 (Attribution of a linear consequence)**.** *Let $G = g_0 + \sum_i g_i u_i$ with independent inputs of variances σ_i^2. Then $\mathrm{Var}[\mathbb{E}[G \mid u_S]] = \sum_{i \in S} g_i^2 \sigma_i^2$ for every subset S of the inputs. The Sobol first-order and total indices of input i are both $g_i^2\sigma_i^2/\mathrm{Var}[G]$, and so is its Shapley share.*

Proof. By independence, $\mathbb{E}[G \mid u_S] = g_0 + \sum_{i \in S} g_i u_i + \sum_{i \notin S} g_i \mathbb{E}[u_i]$, whose variance is $\sum_{i \in S} g_i^2 \sigma_i^2$. The first-order index takes $S = \{i\}$. The total index is one minus the explained fraction of the complement of $\{i\}$, which leaves the same term. In the Shapley average every marginal contribution of input i, $v(S \cup \{i\}) - v(S)$, equals $g_i^2\sigma_i^2$ whatever S is, so the average is that value too. □

On the chain, h_1 has variance 205, of which the source contributes 196, 95.6 percent, and the trunk main 9. For a nonlinear G the same formula with the gradient averaged over the prior, the *activity* $\mathbb{E}[(\partial G/\partial u_i)^2]\,\mathrm{Var}[u_i]$, is a first-order proxy that a census of gradients provides at no cost. It can be wrong, and the three-main problem shows how: the activity shares are 11.7, 76.7 and 11.7 percent for the capacities of mains A, B and C, against Shapley shares of 3.2, 93.6 and 3.2. The parallel mains' capacities matter only when their sum is the binding constraint, which is rarely, and only jointly; the gradient is nonzero whenever they bind and does not see that they

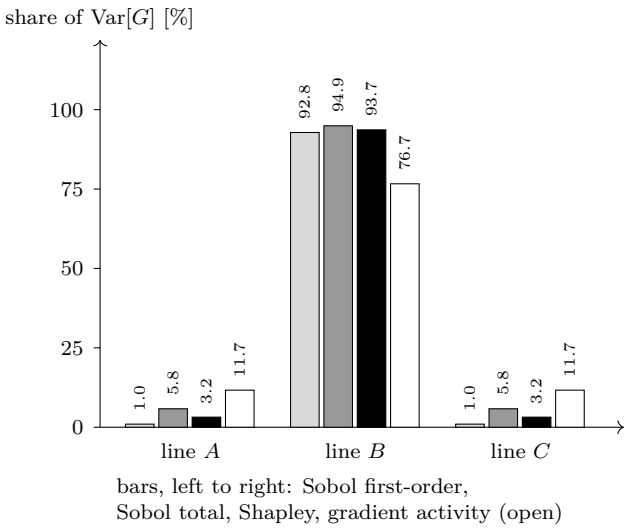


Figure 14.2: Four attributions of the three-main shortfall's variance, in percent: Sobol first-order and total indices and Shapley shares on the 25^3 grid, and the gradient activity from the 300-point census. For mains A and C the Shapley share lies between the first-order and total indices; the activity lies outside them.

bind together. Activity is a screening tool, useful for saying which inputs can be ignored; it is not an attribution.

Figure 14.2 sets four measures side by side, from the script `ch14_more.py`. The Sobol first-order indices of the parallel mains are 1.0 percent each and their totals 5.8: each main matters mostly through its interaction with the other, and 5.2 percent of the variance is interaction, which the first-order indices credit to no one. The Shapley share splits it, 3.2 each, between the first-order and total indices. The activity proxy, 11.7 each, lies outside that bracket: the gradient counts every draw on which the pair binds, as if each main bound alone.

Principle 14.1. *Sensitivity is a derivative at a point; attribution is a decomposition over the prior. The gradient gives the first exactly and the second only for a linear consequence. Where inputs interact, attribution needs conditional expectations, and the question is where to get them cheaply.*

14.4 A census with gradients

Suppose each solve returns G and ∇G at the drawn input, and suppose G is convex in the inputs. Then every solve yields a *supporting plane*, $G(\mathbf{u}) \geq G(\mathbf{u}_j) + \nabla G(\mathbf{u}_j)^{\mathsf{T}}(\mathbf{u} - \mathbf{u}_j)$ for all $\mathbf{u}$, and a census of C solves yields C of them. Their pointwise maximum, floored at whatever lower bound G is known to obey,

$$L(\mathbf{u}) \;=\; \max\Big\{\,0,\; \max_{j \leq C}\big[G(\mathbf{u}_j) + \nabla G(\mathbf{u}_j)^{\mathsf{T}}(\mathbf{u} - \mathbf{u}_j)\big]\Big\} \;\leq\; G(\mathbf{u}) \quad \text{for every } \mathbf{u}, \tag{14.3}$$

is a piecewise-linear convex *surrogate* that lies below G everywhere, not only at the census points and not only under the prior the census was drawn from. This is the cutting-plane construction of convex optimization, used here not to optimize but to bound.

Proposition 14.2 (Supporting planes bound a convex consequence)**.** *Let $G \geq 0$ be convex, and let $\mathbf{s}_j$ be a subgradient of G at $\mathbf{u}_j$ for each census point. Then L of equation* (14.3)*, with $\mathbf{s}_j$ in place of $\nabla G(\mathbf{u}_j)$, satisfies $L \leq G$ everywhere. Consequently, under any distribution of the inputs, $\mathbb{E}[L] \leq \mathbb{E}[G]$ and $\mathbb{P}(L > g) \leq \mathbb{P}(G > g)$ for every g. The surrogate L is convex and piecewise linear, and its maximum over a box is attained at a vertex.*

Proof. A subgradient satisfies $G(\mathbf{u}) \geq G(\mathbf{u}_j) + \mathbf{s}_j^{\mathsf{T}}(\mathbf{u} - \mathbf{u}_j)$ for every $\mathbf{u}$, by definition, and the gradient of a differentiable convex function is one. The maximum of quantities each at most $G(\mathbf{u})$, together with $0 \leq G(\mathbf{u})$, is at most $G(\mathbf{u})$. A pointwise inequality survives any expectation, and the event $\{L > g\}$ is contained in $\{G > g\}$. A maximum of affine functions is convex. A convex function on a box attains its maximum at a vertex, because every point of the box is a convex combination of the vertices and the function there is at most the same combination of the vertex values. □

The inequality is a theorem given convexity. The *certificate* is its empirical check: evaluate G on a holdout set of draws, and verify $L \leq G$ on every one. A violation is not a statistical fluctuation; it is proof that either G is not convex or a gradient is wrong, and either invalidates every bound below.

14.4.1 What the census answers

Once L is built, it is a function that costs nothing to evaluate, and everything asked of G can be asked of L instead, with a known direction of error:

- $\mathbb{E}[L]$ under any belief is a *floor* on $\mathbb{E}[G]$ under that belief, including beliefs other than the one the census was drawn from: a changed forecast, a correlated prior, a what-if on the box.
- $\mathbb{P}(L > g)$ is a floor on $\mathbb{P}(G > g)$: a tail floor at every level.
- $\max L$ over a box is a floor on the worst case, and since L is convex the maximum over a box is at a vertex.
- The mean census gradient is the mean sensitivity, the shadow price of each input's capacity.
- Attribution is computed on L by any method, at zero solves, since L is cheap; where L is close to G the shares are close.

14.4.2 The three-main problem

The shortfall $G = (160 - \min(A + C, B))^+$ is convex in the capacities: the minimum of linear functions is concave, its negative convex, and the positive part preserves convexity. Its gradient at any point is the gradient of the active piece: $(-1, 0, -1)$ when the parallel pair binds, $(0, -1, 0)$ when main B does, zero when nothing is short. A census of 300 draws with these gradients gives the surrogate of equation (14.3). Table 14.1 certifies it on 20,000 holdout draws and answers the questions.

On this problem the surrogate is not merely a floor; it is exact, and the reason is instructive. The shortfall has three linear pieces, and once the census has landed at least one point on each, the three supporting planes reproduce the function everywhere. The gap between L and G on the holdout is zero to machine precision, the floors are the values, and the Shapley shares computed on a midpoint grid of L, at zero further evaluations of G, are the reference shares of Chapter 10 to a tenth of a point. On a real load-shedding model, a linear program whose value has as many pieces as the program has bases, the gap is finite and the floors are floors; a companion census

Table 14.1: A census of 300 evaluations with active-set gradients on the three-main shortfall: the certificate on 20,000 holdout draws and the answers read from the surrogate at zero further evaluations, against the truth.

question	from the surrogate L	truth	direction
certificate: $\max(L - G)$ on holdout	3×10^{-14}	≤ 0 required	—
$\mathbb{E}[G]$ under the census belief	5.322	5.333 (holdout 5.322)	floor
$\mathbb{E}[G]$, A, C shifted to $[60, 110]$	8.274	8.274	floor
worst case over the box	20.0	20.0 at $(70, 140, 70)$	floor
$\mathbb{P}(G > 5)$, $\mathbb{P}(G > 10)$, $\mathbb{P}(G > 15)$	0.400, 0.266, 0.131	same	floors
shadow prices $\mathbb{E}[\nabla G]$	$(-0.05, -0.51, -0.05)$	—	—
Shapley shares on a 25^3 midpoint grid of L	3.2, 93.7, 3.2	3.2, 93.7, 3.2	—

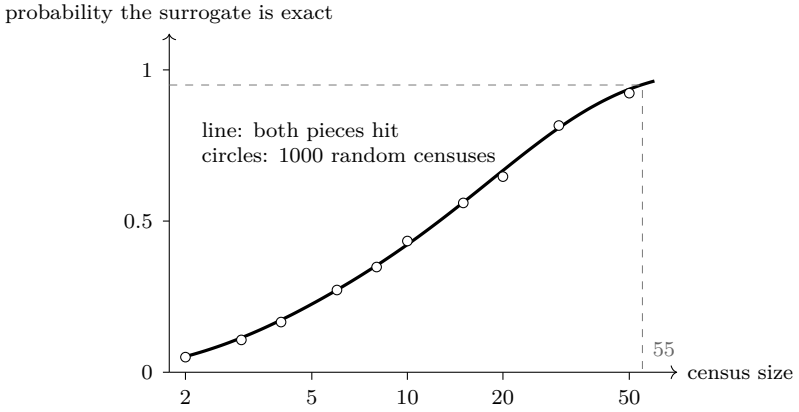


Figure 14.3: The probability that a census of random draws gives an exact surrogate of the three-main shortfall, against the census size. Line: the probability that the census hits both curtailed pieces. Circles: the fraction of 1,000 random censuses whose surrogate is exact on the holdout. The dashed line marks 95 percent, reached at 55 points.

study in the power domain, committed to the repository that accompanies the book, runs exactly this construction on a 73-bus network with fourteen uncertain ratings and nine questions, and its solve counts were the ones Chapter 10 quoted: 6,536 against 113,876.

The floors under a shifted belief deserve a second look. The census was drawn with the parallel mains' capacities on $[70, 120]$; the question asked what the shortfall would be if they were on $[60, 110]$, a worse world. No new evaluation of G was made. The surrogate, being a bound everywhere, is a bound there too, and in this case it is the answer. A sampler would have drawn a new sample.

How many census points does exactness take? The surrogate is exact once the census has a point on each curtailed piece, the one where the parallel pair binds and the one where main B binds; the zero piece is the floor. A draw lands on the first with probability 0.054 and on the second with 0.486, so a census of C random points is exact with probability $1 - (1 - 0.054)^C - (1 - 0.486)^C + (1 - 0.054 - 0.486)^C$. Figure 14.3 draws it against 1,000 random censuses at each size. The rare piece sets the size: 55 points for 95 percent and 84 for 99, against two placed deliberately, one on each piece. A census drawn from the prior spends its points where the prior puts its mass, and a rare regime needs either many points or one placed on purpose.

Table 14.2: The certificate with active-set gradients and with central finite differences of G (step 1 L/s) on the same 300 census points and 20,000 holdout draws.

gradient	census points at a kink	holdout violations	$\max(L - G)$
active set	0 of 300	0 of 20,000	3×10^{-14} L/s
central differences	18 of 300	984 of 20,000	+0.124 L/s (1.9% of sd)

Principle 14.2. *A convex consequence with gradients from the solve yields supporting planes, and their maximum is a surrogate that bounds the consequence everywhere. Every question asked of the surrogate is answered at zero further solves with a known direction of error, and the certificate is a holdout check that fails loudly.*

14.5 The kink

Everything in the previous section rested on the gradient being a subgradient: a slope that supports the function from below. For a convex function that is smooth the gradient is the only subgradient and any correct computation of it will do. For a convex function with a kink, the minimum of the three-main problem, the positive part, the maximum in a limiter, the active-set gradient is a subgradient and the finite difference is not.

Replace the active-set gradients of the census by central differences of G with a step of one litre a second, and re-certify. Table 14.2 gives the result. Eighteen of the three hundred census points straddle a kink, in the sense that the difference step crosses from one linear piece to another; at each of them the central difference averages the two slopes and the resulting plane crosses the kink and sits above G nearby. On the holdout the surrogate exceeds G at 984 of 20,000 points, by up to 0.12 litres a second, two percent of the shortfall's spread. The certificate fails. Nothing about the function has changed; only the gradient, and only where the function bends.

Proposition 14.3 (A central difference across a kink)**.** *Let f be convex and piecewise linear in one variable, with a single kink at which its slope rises by J. A central difference with step h, taken at a centre within h of the kink, gives a line that lies above f at the kink by up to $Jh/8$, the largest value reached when the centre is $h/2$ from the kink. A central difference whose centre is farther than h from every kink returns the slope of its piece, which is a subgradient.*

Proof. Put the kink at 0 and subtract the left-hand line, which changes neither the difference quotient's error nor the violation, so that $f(x) = J\,x^+$. For a centre $x_0 \in (-h, h)$ the central difference is $s = (f(x_0 + h) - f(x_0 - h))/(2h) = J(x_0 + h)/(2h)$. The line through $(x_0, f(x_0))$ with slope s has height $f(x_0) - s\,x_0$ at the kink, where f is zero. For $x_0 < 0$ that height is $-Jx_0(x_0 + h)/(2h)$, and for $x_0 > 0$ it is $Jx_0(h - x_0)/(2h)$. Both are positive, and each is largest, $Jh/8$, at $|x_0| = h/2$. A centre farther than h from the kink differences a single linear piece and returns its slope. □

The census violation of 0.124 litres a second in Table 14.2 matches this bound: the slope along the capacity of main B jumps by one where B takes over from the parallel pair, and the step was one litre a second, so $Jh/8 = 0.125$. Figure 14.4 draws it on a line through the kink. In several variables the bound is only a lower estimate. Each coordinate's difference errs by its own

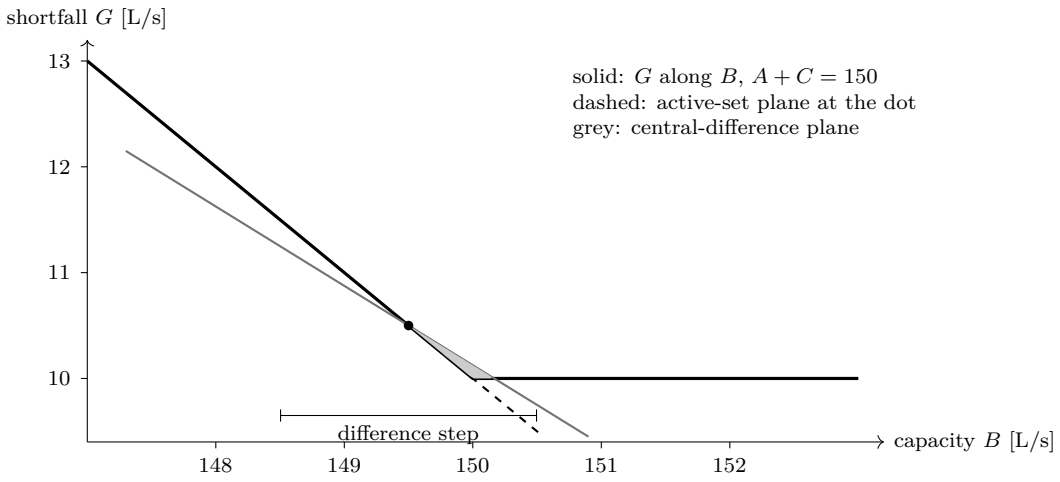


Figure 14.4: The three-main shortfall along the capacity of main B with $A + C = 150$ (solid), which has a kink at $B = 150$ where its slope rises from -1 to 0. At the dot, 149.5, the active-set plane (dashed) has the slope of its piece and touches G without crossing it. The central-difference plane (grey), whose step of one litre a second reaches across the kink, has slope -0.75 and lies above G by up to 0.125 litres a second, the shaded sliver: an eighth of the step times the jump in slope.

amount, so the plane is tilted along the kink as well as across it, and along the kink the violation grows with distance; on the overload of Exercise 14.5 it reaches fifteen times the one-dimensional bound.

The fix is not a smaller step; a smaller step straddles fewer points and each straddled point is just as wrong. The fix is to difference what is smooth and compose with what is not: the flows and heads that the physics solves for are smooth in the inputs, and the kink is in the consequence built from them, a minimum, a positive part, a limit exceeded; take the gradient of the smooth inner map, by equation (14.1) or by differences, and apply the active piece of the outer function at the center point. That is the active-set gradient of Table 14.1, it costs no extra solves, and it is what Chapter 7 meant by differentiating within the regime. The rule is the one that a finite-difference adapter must never break: it may return a gradient, but it must never declare the consequence convex, because its gradients across kinks are not subgradients and the certificate they produce is worthless.

Principle 14.3. *A finite difference across a kink is not a subgradient, even of a convex function. Difference the smooth inner map and apply the active set at the center. Never let a differenced gradient carry a certificate.*

14.6 Worked example: convexity lost and recovered

The two treatment works of Chapter 10 test the census on a second network. Their unserved demand $U = (2{,}100 - S_1 - S_2)^+$, with $S_k = \min\big((1-\alpha D)\sum_j R_{kj},\ M_k\big)$, has an active-set gradient everywhere, by the same reasoning as the restriction's. On the treatment-limited piece a works' delivery rises with each stream's output at the derated rate and falls with the drawdown at α

Table 14.3: Attribution of the two treatment works' shortfall variance to three groups of inputs, in percent: Sobol first-order and total indices and Shapley shares by pick-freeze on 400,000 draws per sample set, and the gradient activity summed over each group.

group	first-order	total	Shapley	gradient activity
the river	36.6	76.9	55.9	71.0
works 1	10.2	35.8	22.2	14.5
works 2	10.2	35.4	22.0	14.5

times the output sum; on the transfer-limited piece it rises with the main's capacity. A census of 500 draws over all nine inputs builds the surrogate, and the holdout refuses to certify it: 3,863 of 20,000 holdout points lie below the surrogate, by up to 5.5 litres a second, a fifth of the shortfall's spread. Figure 14.6 shows where. The violations sit at deep drawdowns, 8.4 centimetres on average.

The gradients are right; the function is not convex. The derated output $(1-\alpha D)R$ is bilinear in the drawdown and a stream's rating, and its Hessian in those two inputs has eigenvalues $\pm\alpha$: a saddle. The shortfall inherits the saddle wherever the treatment streams bind, which at deep drawdowns is almost everywhere.

The cure is the separator of Chapter 10. At a fixed drawdown the derating is a number, each works' output is linear in its stream ratings, its delivery is the minimum of two linear functions and so concave, and the shortfall, the positive part of a constant minus two concave functions, is convex in the other eight inputs. Stratify the census by the drawdown: at each of nine drawdown nodes, 120 census points in the other eight inputs and 5,000 holdout points. The certificate holds at every node, with a worst violation of 10^{-12} litres a second, rounding, in 45,000 holdout points. The floors on the conditional mean are exact at the four deepest nodes, 3.91, 14.23, 33.02 and 64.81 litres a second. At 5 centimetres the floor is 0.92 against 1.03, and at the two shallower shortfall nodes it is zero, because the census found little or none of the small shortfall region there.

Attribution by groups. The nine inputs fall into three groups an operator would name: the river, works 1 with its three streams and its transfer main, and works 2. Table 14.3 gives four measures for the groups, from seven sample sets of 400,000 draws each by pick-freeze; the works' model costs almost nothing to evaluate, so the sampling route is affordable here. The river's first-order index is 37 percent, the share of the shortfall's variance that the drawdown alone explains; the nine-node grid of Chapter 10 put it at 39. Each works' is 10. The first-order indices sum to 57 percent, so 43 percent of the variance is interaction: a shortfall needs a deep drawdown and a weak works together. The totals, 77, 36 and 35, count the interaction in full for every group that takes part in it. Shapley splits it, 56 percent to the river and 22 to each works. The gradient activity says 71 and 15. It weighs each input by its slope where the shortfall is positive, and the drawdown's slope, the derating of six streams at once, is large wherever there is a shortfall. It does not see that the shortfall exists only when a works is also weak, which is where the works earn their Shapley share.

A floor under a forecast. The stratified census is a set of surrogates, one per drawdown, and it can answer questions about any belief over the drawdown at no further cost, provided the

step between drawdown nodes is bounded too. Extend it to 29 nodes from 6 to 13 centimetres, a quarter centimetre apart, with 120 census points each: 3,480 evaluations in all. The census floors the conditional mean at each node. Between nodes the physics supplies the rest of the argument. The shortfall never falls as the drawdown deepens, since a lower river derates every stream and changes nothing else, so on each interval between nodes the conditional mean is at least its value at the lower node. Weighting each node's floor by the forecast's probability of the interval above it therefore gives a certified floor, a lower Riemann sum.

Proposition 14.4 (A floor under any belief over a monotone input)**.** *Let $G(t, \mathbf{x}) \geq 0$ be nondecreasing in a scalar input t for every $\mathbf{x}$, let $t_1 < \cdots < t_m$ be nodes, and let $L_i(\mathbf{x}) \leq G(t_i, \mathbf{x})$ for each i. For any distribution of t independent of $\mathbf{x}$, with $t_{m+1} = \infty$,*

$$\mathbb{E}[G] \;\geq\; \sum_{i=1}^{m} \mathbb{P}(t_i \leq t < t_{i+1})\, \mathbb{E}[L_i(\mathbf{x})]. \tag{14.4}$$

Proof. For t in $[t_i, t_{i+1})$, monotonicity gives $G(t, \mathbf{x}) \geq G(t_i, \mathbf{x}) \geq L_i(\mathbf{x})$, and for $t < t_1$, $G(t, \mathbf{x}) \geq 0$. Take the expectation over $\mathbf{x}$ with t fixed, which by independence does not change the distribution of $\mathbf{x}$, and then over t, splitting its range at the nodes. □

Under the drought forecast of Chapter 13, 8.5 ± 0.7 centimetres, it is 28.18 litres a second, against 30.58 ± 0.09 from 200,000 direct draws. Under a deeper forecast of 9.5 ± 0.7 centimetres it is 49.53, against 53.05 ± 0.11. Figure 14.5 sweeps the forecast.

The floor is loose by roughly half a node spacing times the slope of the conditional mean, two or three litres a second here, and halving the spacing halves it, at twice the census. The tempting alternative is to weight each node by the forecast's density instead. That gives 30.44 under the drought, much closer, but it is not a bound: under a shallower forecast of 7 ± 0.7 centimetres it gives 10.55 litres a second where direct draws give 10.14 ± 0.05. The conditional mean is convex in the drawdown, and a density-weighted sum over nodes overstates a convex function. A floor at each node becomes a floor over the forecast only through an argument about what happens between the nodes, and here monotonicity is that argument. Each new forecast costs no evaluation of the works at all.

> **Principle 14.4.** *Convexity is a property of a consequence in chosen inputs. When one input spoils it, condition on that input and certify the rest; the input to condition on is usually a separator already.*

14.7 Conditions

The chapter's method rests on two conditions, and it is worth stating where each holds.

The solve must return its gradient. Equation (14.1) does this for any model whose residuals are differentiable in the inputs, at the cost of keeping the input columns of the Jacobian. A load-shedding linear program returns its gradient as the dual variables on the rating constraints, for free. An iterative solver that returns only a value must be differenced, with the caveat of §14.5.

The consequence must be convex in the inputs. The value of a linear program is convex in its right-hand sides, so any consequence that is the optimal value of a linear allocation, load shed, redispatch cost, is convex in the capacities and loads that bound it. A DC power flow's line

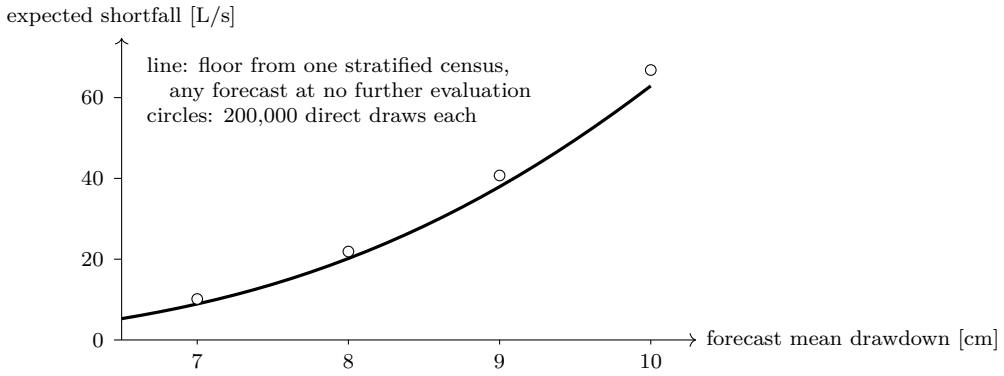


Figure 14.5: The expected unserved demand of the two treatment works under a drawdown forecast of mean μ and spread 0.7 centimetres, as μ sweeps from 6.5 to 10 centimetres. Line: the certified floor from one stratified census of 3,480 evaluations, a lower sum over drawdown nodes that uses the shortfall's monotonicity in the drawdown; every forecast costs no further evaluation. Circles: 200,000 direct draws at each of four forecasts.

flows are linear in the injections, so any convex function of them, an overload measured as a positive part, is convex. An AC power flow is not linear, its flows are not convex in the injections, and a surrogate built on it is a surrogate without a certificate: still useful for screening, mean sensitivities and variance reduction, none of which need the bound, but no longer a floor. The same is true of a water network: head loss is quadratic in flow, so the heads are not convex in the demands, and a surrogate built on a solved hydraulic model screens and attributes but certifies nothing. A companion study in the power domain, committed to the repository that accompanies the book, puts the nine questions to an AC model, the IEEE 14-bus network with eight uncertain loads and total line overload as the consequence, in a congested regime and an intermittent one. The shadow prices, the total Sobol indices and the first-order indices survive as estimates in both. The screening bound stays certified in both, because it never needed the inequality. The five questions that read a floor from the surrogate, a new belief, correlated loads, a what-if curve, the tail floors and the worst case, lose the certificate. In the congested regime they are empirical: $L \le G$ held on all 2,000 holdout and 10,000 reference draws, but nothing guarantees that it holds elsewhere. In the intermittent regime they are estimates: $L \le G$ fails on 1,149 of the 10,000 reference draws, by at most 7×10^{-4} of the standard deviation of G.

14.8 In practice

Take gradients from the solve. Keep the input columns of the Jacobian when assembling it. For one consequence and many inputs, solve the adjoint system once; for one input and many consequences, solve forward once. Difference only what the solver cannot differentiate, and only the smooth part of it.

Screen with the gradient, attribute with conditional expectations. The activity proxy says which inputs can be dropped. The shares that go into a report come from Shapley or Sobol on a grid or on a certified surrogate, where they cost no further solves.

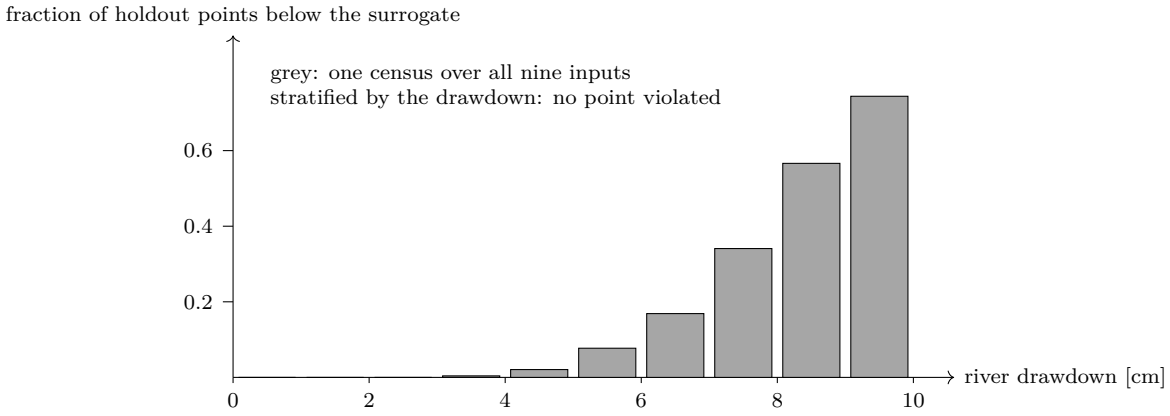


Figure 14.6: The two treatment works' shortfall and a census with active-set gradients over all nine inputs: the fraction of holdout points at which the surrogate exceeds the shortfall, by drawdown bin. The violations are at deep drawdowns, where the derating's saddle matters. Stratified by the drawdown, the same construction violates at no holdout point.

Certify before using any floor. Check $L \leq G$ on a holdout drawn independently of the census. One violation invalidates every bound built from that surrogate, and it points at either a wrong gradient or a lost convexity.

Place census points on purpose. A census drawn from the prior finds a rare regime late: 55 random points for what two placed points do on the three-main problem. Seed the census at the vertices of the box and at the boundaries of the regimes you know about.

Turn floors at nodes into a floor over a belief with an argument. A surrogate stratified on one input bounds the consequence at each node and nowhere between. A monotone input gives the argument that carries the bound between nodes, Proposition 14.4; without one, a density-weighted sum over the nodes is an estimate, and it can overstate.

Check convexity in the coordinates you use. A product of inputs, a derating, an efficiency that varies with load: each can make a consequence non-convex. Stratify on the input that causes it, and certify within each stratum.

14.9 What is missing

Every input in this chapter was continuous. The inputs that fail discretely, the availabilities of Chapter 7, have no gradient, and the question of which of them matter most, alone and together, needs the branch machinery instead: which combinations of failures are most probable, which most damaging, and how a common cause ties them together. That is Chapter 15, the last of Part IV.

Notes and further reading

Sensitivities by the implicit function theorem and their adjoint form are standard in design optimization and in data assimilation; Nocedal and Wright [65] treat the linear algebra, and the observation that a linear program's duals are the derivatives of its value with respect to its right-hand sides is the sensitivity theorem of linear programming, found there and in Boyd and Vandenberghe [11], who also give the convexity of the optimal value in the constraint data that §14.7 relies on. Variance-based attribution is Sobol's [85]; the Shapley share as an attribution that resolves interactions is Owen's [67], with estimation by Song, Nelson and Staum [86]; the pick-freeze estimator is Saltelli's [81]. The gradient-based activity as a screening proxy is common in sensitivity analysis under various names; its failure on the three-main problem is the book's illustration.

Supporting planes of a convex function and their pointwise maximum are the cutting-plane method of Kelley [43]; the use of that construction as a certified surrogate, evaluated under beliefs other than the census's and checked on a holdout, is the framework's, and the case study cited in §14.4 is its first full instance. The kink caveat of §14.5 was found the hard way, on an AC model whose consequence was a sum of positive parts. In the AC part of the surrogate case study, in its intermittent regime, differenced gradients produced certificate violations of 1.09 litres a second, about a sixth of the consequence's spread of 6.38. The active-set correction cut them by more than two orders of magnitude, to 0.0046 litres a second; the residue, under a thousandth of the spread, is the model's non-convexity itself. The caveat is recorded here so that no reader repeats it.

Summary

- Sensitivity, attribution and certification all start from $\partial G/\partial \mathbf{u}$, which the implicit function theorem gives from the two Jacobians the solve assembled: n_u forward solves or one adjoint solve. On a 100×100 grid network one adjoint solve, 1.2 milliseconds, gives the centre head's sensitivity to all ten thousand sources, against 23 seconds forward.
- Attribution is a variance decomposition over the prior; Sobol indices and Shapley shares define it; the gradient gives it exactly only for a linear consequence. On the three-main problem the gradient proxy says 12/77/12 and the truth is 3/94/3, with the parallel mains' Shapley shares between their first-order and total Sobol indices.
- A convex consequence with gradients yields supporting planes whose maximum bounds it everywhere. On the three-main problem 300 solves give an exact surrogate; floors on the mean, the tail, the worst case and a shifted belief, and the Shapley shares, follow at zero further solves.
- A finite difference across a kink is not a subgradient: 18 straddled census points broke the certificate at 984 holdout draws, by up to $Jh/8$, an eighth of the step times the jump in slope. Difference the smooth inner map and apply the active set at the center.
- The surrogate is exact once the census hits every piece; a random census needs 55 points for that with probability 0.95, two placed points always suffice.
- On the two treatment works a census over all nine inputs fails its certificate at deep drawdowns, because the derating is bilinear; stratified by the drawdown it holds at every holdout point. With the shortfall's monotonicity in the drawdown it floors the expected shortfall under any forecast at no further evaluations: 28.2 litres a second in the drought,

against 30.6 by direct sampling. A density-weighted sum over the nodes is closer but is not a bound.

- Over three groups the works' shortfall is 56 percent river and 22 percent each works by Shapley. 43 percent of its variance is interaction, and the gradient proxy overstates the river at 71.
- Conditions: the gradient must come from the solve, and the consequence must be convex. Linear programs and DC flows qualify; AC flows do not, and lose the bound but not the screening.

Exercises

Exercise 14.1. Derive equation (14.1) by differentiating $\mathbf{F}(X(\mathbf{u}), \mathbf{u}) = \mathbf{0}$, and derive the adjoint form for $G = \mathbf{c}^{\mathsf{T}}\mathbf{z}$. Count the solves for n_u inputs and n_c consequences in each form and state when each is preferable.

Exercise 14.2. For the three-main problem, compute the Sobol first-order and total indices of the three ratings on the 25^3 grid, and show that the totals exceed the first-order indices for mains A and C. Relate the difference to the interaction the activity proxy missed.

Exercise 14.3. Prove that the positive part of a convex function is convex, that the minimum of two linear functions is concave, and hence that the three-main shortfall is convex. Then give a two-main scheme whose shortfall is not convex in its capacities.

Exercise 14.4. Reduce the census of Table 14.1 to 30 points and re-certify. Find the smallest census for which the surrogate is still exact on the holdout, and explain the number in terms of the pieces of G.

Exercise 14.5. Implement the kink fix for the three-main problem with finite differences: difference the smooth quantities $A + C$ and B (trivially) and apply the active piece of G at the center, and show the certificate holds. Then apply the same construction to the water loop with an overload consequence $\max(0, |Q_{12}| - 1.4)$ and uncertain demands, differencing the flows and applying the active set.

Solutions to selected exercises

Exercise 14.1. The derivation is given after equation (14.1). For n_u inputs and n_c consequences the forward form needs n_u solves, each giving one column of $\partial X/\partial \mathbf{u}$ and hence the sensitivity of every consequence to that input. The adjoint form needs n_c solves, each giving the gradient of one consequence with respect to every input. With the factorization held, each solve is two triangular solves, so the cheaper form is the one with the smaller count: adjoint when there are fewer consequences than inputs, forward otherwise. On the chain the body head's sensitivities, $(0.700, 1.000, -4.000)$ to the source, the trunk main and the conductance, come from one adjoint solve with $\mathbf{c}$ the unit vector on h_1.

Exercise 14.2. On the 25^3 grid the variance is 41.92. The first-order indices are 1.0, 92.8 and 1.0 percent for mains A, B and C; the totals are 5.8, 94.9 and 5.8. The first-order indices sum to 94.8 percent, so 5.2 percent of the variance is interaction, and for each parallel main the total

exceeds the first-order index by 4.8 points: almost all of what the parallel mains contribute, they contribute jointly. Main B's total exceeds its first-order index by only 2.1 points. The activity proxy missed exactly this: the gradient $(-1, 0, -1)$ on the piece where the pair binds credits each main with the full slope whenever the pair binds, as though either alone could be varied to relieve the shortfall, which is the interaction counted twice.

Exercise 14.4. A 30-point census still certifies, as every census with active-set gradients must, since its planes are subgradients; what it may lose is exactness. It is exact with probability 0.81 (Figure 14.3), and when it is not, its mean gap is small, 0.06 litres a second on average over random 30-point censuses. The smallest census that is exact is two points, one on the piece where the parallel pair binds and one on the piece where main B binds, since the shortfall has three linear pieces and the floor at zero supplies the third. Drawn at random, the rare piece, hit with probability 0.054, sets the size: 55 points for 95 percent.

Exercise 14.3. If f is convex, $\max(f, 0)$ is the maximum of two convex functions. A maximum of convex functions is convex, because at a convex combination of points each function is at most the same combination of its values, and so is their maximum. The minimum of two linear functions is concave by the same argument with the inequality reversed, and its negative is convex. The shortfall is the positive part of 160 minus that minimum, so it is convex. For a two-main scheme whose shortfall is not convex, feed a demand of D through a changeover valve from either of two mains, one at a time, whichever has the larger rating. The deliverable power is $\max(A, B)$, which is convex, and the shortfall $(D - \max(A, B))^+$ is the positive part of a concave function, which need not be convex: on the segment from $(A, B) = (D, 0)$ to $(0, D)$ it is zero at both ends and $D/2$ in the middle. A census on it would produce planes that cut above it, and the certificate would say so.

Exercise 14.5. On the water loop with equal link conductances the flow on link 1–2 is linear in the demands, $\partial Q_{12}/\partial(d_2, d_3) = (-0.667, -0.333)$, and the overload $\max(0, |Q_{12}| - 1.4)$ is convex, positive on 19.8 percent of the box $d_2 \in [-2, -1]$, $d_3 \in [-1, -0.2]$. With central differences of the overload, step 0.05 kg/s, 18 of 200 census points straddle the kink, and the surrogate exceeds the overload at 7,445 of 20,000 holdout points, by up to 0.065 kg/s. That is fifteen times the one-dimensional bound $Jh/8 = 0.0042$ along d_2: each coordinate's difference errs differently, the plane tilts along the kink line, and the violation grows with distance along it. Differencing the flow instead, which is exact here because the flow is linear, and applying the active piece of the overload at the centre gives no violations, the largest gap being 2.6×10^{-15}. The three-main construction of the exercise's first part is the same: difference $A + C$ and B, which are the inputs themselves, and apply the active piece.

Chapter 15

Combinatorial failure

The interventions of Chapter 13 were chosen by the operator. The failures of this chapter are not. Lines trip, pumps stop, valves stick, and they do so in combinations that nobody selected, at rates that depend on the weather, and sometimes for a shared reason. The operator's questions are then combinatorial: which components are most likely to fail together, which combinations would hurt most, how much of the total risk the plausible combinations carry, and what it costs to leave the rest unexamined. This chapter treats those questions as inference on the hybrid model of Chapter 7, one branch per failure set, and shows that the machinery already built answers them: a prior over failure sets that the pruning proposition of Chapter 9 makes enumerable, a consequence per branch that is an intervention of Chapter 13 and a rank-one update of Chapter 11, and a mixture over the branches whose weights and tails are the risk. Two worked examples carry it, a water network of seven pipes and a booster station with four pumps, and the second shows what the first cannot: that a shared cause defeats redundancy in a way independent failures never do.

15.1 The question

A system with L components that can fail has 2^L possible failure sets. Let $S \subseteq \{1, \dots, L\}$ be the set of failed components, let $\mathbb{P}(S)$ be its prior probability, and let $C(S) \geq 0$ be the consequence of that failure set: the load that cannot be served, the overload beyond ratings, the number of metres a room exceeds its limit, or whatever the operator counts. The four questions are:

1. *Risk*: what is $\mathbb{E}[C] = \sum_S \mathbb{P}(S)\, C(S)$, and what is the probability $\mathbb{P}(C > c)$ of a consequence beyond some level?
2. *Ranking*: which failure sets contribute most to the risk, $\mathbb{P}(S)\, C(S)$ in decreasing order?
3. *Worst case*: among sets of size k, which has the largest consequence, $\max_{|S|=k} C(S)$? This is the $N - k$ question in its deterministic form, and the set that attains it is the one an adversary would choose.
4. *Coverage*: if only sets up to size k are examined, how much of the risk has been captured, and how much probability has been left unexamined?

The deterministic tradition answers only the third, usually for $k = 1$: a system is $N - 1$ *secure* if no single failure produces an unacceptable consequence, and the check is to try each of the L single failures in turn. That is a valuable check and an incomplete one. It weights every single failure equally, though some are far more likely than others; it says nothing about pairs, though pairs happen; and it reports a verdict, secure or not, rather than a risk. The probabilistic

Table 15.1: Coverage by enumeration depth under independent equal-rate failures: the number of failure sets of size exactly k, the probability that at most k components fail, and the probability left unexamined.

	$L=7$, $q=0.02$			$L=100$, $q=0.01$			$L=500$, $q=0.005$		
k	sets	$\mathbb{P}(\lvert S\rvert \le k)$	left	sets	$\mathbb{P}(\lvert S\rvert \le k)$	left	sets	$\mathbb{P}(\lvert S\rvert \le k)$	left
0	1	0.868	0.13	1	0.366	0.63	1	0.082	0.92
1	7	0.992	$7.9\,10^{-3}$	100	0.736	0.26	500	0.287	0.71
2	21	0.9997	$2.6\,10^{-4}$	4,950	0.921	0.079	$1.2\,10^{5}$	0.543	0.46
3	35	$1-5\,10^{-6}$	$5.3\,10^{-6}$	$1.6\,10^{5}$	0.982	0.018	$2.1\,10^{7}$	0.758	0.24
4	35	$1-7\,10^{-8}$	$6.5\,10^{-8}$	$3.9\,10^{6}$	0.997	$3.4\,10^{-3}$	$2.6\,10^{9}$	0.892	0.11
5	21	1	$4\,10^{-10}$	$7.5\,10^{7}$	0.9995	$5.4\,10^{-4}$	$2.6\,10^{11}$	0.958	0.042

version answers all four, and the $N-1$ check is its $k=1$ slice.

15.2 The prior over failure sets

15.2.1 Independent failures

The simplest prior takes each component to fail independently with its own probability q_i over the horizon of interest. Then

$$\mathbb{P}(S) \;=\; \prod_{i\in S} q_i \prod_{i\notin S}(1-q_i), \tag{15.1}$$

and when the rates are equal, $q_i = q$, the probability depends only on the size of the set: $\mathbb{P}(S) = q^{|S|}(1-q)^{L-|S|}$, and the number of failures is binomial with mean Lq. The prior decays geometrically in the size of the set, by a factor $q/(1-q)$ per additional failure, and this decay is what makes the combinatorial question tractable.

Proposition 15.1 (Coverage by size)**.** *Under independent equal-rate failures, the total probability of all failure sets of size greater than k is the binomial tail $\mathbb{P}(|S|>k) = \sum_{j>k}\binom{L}{j}q^j(1-q)^{L-j}$. By Proposition 9.3, examining only the sets of size at most k and renormalizing errs in any probability by at most that tail; the risk it misses is at most the tail times the largest consequence.*

Proof. The first statement is the definition of the binomial distribution applied to $|S|$, which is a sum of L independent Bernoulli variables with success probability q. The second is Proposition 9.3 with the discarded set being every S with $|S|>k$ and the discarded weight being its total probability. For the risk, $\sum_{|S|>k}\mathbb{P}(S)C(S) \le \max_S C(S)\sum_{|S|>k}\mathbb{P}(S)$. □

Table 15.1 evaluates the tail for three systems, from the script `ch15_combinatorial.py` in the book's `scripts` directory, which supplies every number in the chapter. Seven lines at two percent, the small network below, are covered to 2.6×10^{-4} by 29 sets. A hundred lines at one percent, a real transmission region, need the 161,700 sets of size three to reach 98 percent and the 3.9 million of size four to reach 99.7. Five hundred components at half a percent, a large network or a data center's equipment list, are not covered to 96 percent until size five, at 2.6×10^{11} sets, which no one enumerates. The decay that makes seven lines trivial makes five hundred hard, because the expected number of failures, Lq, has grown from 0.14 to 2.5, and the sets that carry the probability are the ones with two or three failures, of which there are millions.

Three things follow from the table. On a small system, enumerate everything; there is no reason not to. On a medium system, enumerate to the depth the tail requires for the accuracy required, which for a risk accurate to one percent is $k = 3$ at a hundred lines, and report the tail as the unexamined probability. On a large system, enumeration to sufficient depth is impossible, and one of three things must give: the consequence must be cheap enough to sample the prior instead of enumerating it, the structure must separate the components into regions whose failure sets can be enumerated independently, or the question must be narrowed from "all sets" to "the sets that matter", which is the ranking and worst-case questions with bounds. The rest of the chapter takes the first two systems and the last option.

15.2.2 Failure as an intervention

A failed component is an intervention in the sense of Chapter 13: the closure of a failed line is replaced by "no flow", the closure of a failed pump by "no heat removed", and in the hybrid model of Chapter 7 the replacement is carried by an availability variable a_i in the closure's scope, $F_{ij} = a_{ij} B(\theta_i - \theta_j)$. A failure set S is the branch with $a_i = 0$ for $i \in S$ and $a_i = 1$ otherwise. The consequence $C(S)$ is a readout of that branch: the overload of the surviving lines, the load at buses the failures have cut off, the duty of a station whose pumps are down. The prior $\mathbb{P}(S)$ is the product of the availability priors. And the risk $\mathbb{E}[C]$ is the mixture of the branch consequences by their weights, which is what Chapter 7 said a discrete variable does to a posterior, now with L of them.

15.3 The consequence per branch, and how to get it cheaply

For a network in the DC approximation, the consequence of a failure set needs the flows with the failed lines removed, and two things can happen. The surviving network can remain connected to the reference, in which case the flows redistribute and some may exceed their ratings; or a failure can cut a set of buses off entirely, an *island*, in which case those buses can be served by nothing and their load is shed. Both are visible from the graph before any solve: connectivity from the reference is a graph search, and the islanded buses' load is a sum.

For the connected part, the flows after an outage are a rank-one update of the flows before it, and the update has a name.

Proposition 15.2 (Line outage distribution factors)**.** *In a DC network, let F_e be the pre-outage flow on line e, and let line o be removed without islanding any bus. Then the post-outage flows are*

$$F'_e \;=\; F_e + \mathrm{LODF}_{e,o}\, F_o, \qquad \mathrm{LODF}_{e,o} \;=\; \frac{\mathrm{PTDF}_{e,o}}{1 - \mathrm{PTDF}_{o,o}}, \tag{15.2}$$

where $\mathrm{PTDF}_{e,o}$ is the change in the flow on line e when one unit of power is injected at the sending end of line o and withdrawn at its receiving end, with all lines in service.

Proof. Write the nodal equations of Chapter 2 as $\mathbf{B}'\boldsymbol{\theta} = \mathbf{P}$, with $\mathbf{B}'$ the reduced susceptance matrix, and let $\mathbf{a}_o$ be the incidence column of line o (entries $+1$ at its sending bus, -1 at its receiving bus, zero elsewhere, reference row dropped). Removing line o subtracts its contribution: $\mathbf{B}'' = \mathbf{B}' - B_o \mathbf{a}_o \mathbf{a}_o^{\mathsf{T}}$, a rank-one change. Rather than invert $\mathbf{B}''$, observe that the network with line o removed and the same injections carries the same flows as the network with line o present, the same injections, and an additional transfer of F'_o units injected at o's receiving end and

withdrawn at its sending end, chosen so that line o carries zero net flow. (The transfer cancels the line's flow, and a line carrying zero flow can be removed without changing anything else.) With line o present the flows are linear in the injections, so the transfer of t units along o's path changes every flow by $t\,\mathrm{PTDF}_{e,o}$, by definition of the factor. The line's own flow becomes $F_o + t\,\mathrm{PTDF}_{o,o}$, and setting the net flow to zero after subtracting the transfer itself, $F_o + t\,\mathrm{PTDF}_{o,o} - t = 0$, gives $t = F_o/(1 - \mathrm{PTDF}_{o,o})$. Every other flow is then $F_e + t\,\mathrm{PTDF}_{e,o}$, which is equation (15.2). The denominator vanishes only when $\mathrm{PTDF}_{o,o} = 1$, which is when line o is the only path between its ends, which is when removing it islands, the case excluded. □

The proposition is the Sherman–Morrison identity of Chapter 11 in physical clothing: a line outage is a rank-one change to the precision of the network, and its effect on every flow is one column of the pre-outage sensitivity, scaled. The column $\mathrm{PTDF}_{\cdot,o}$ is one solve against the base-case factorization, so the consequences of all L single outages cost L solves against one factorization, and pairs cost two applications of the same formula. On a six-bus DC network the script checks the proposition on the outage of line 1–3: the predicted flows agree with a full re-solve to 5×10^{-15}.

The proposition is stated for a linear network and it is used here on one, which is why the example that checks it is electrical rather than hydraulic. A water network has no such shortcut. Head loss goes as $Q^{1.852}$, so the flows are not linear in the demands, there is no matrix whose column gives the redistribution after an outage, and the consequence of each failure set is a fresh Newton solve. What survives is the warm start: the base-case state is a good starting point for a single outage, and §15.4's 128 solves each converge in a handful of iterations from it. Cheapness by superposition is a privilege of linear domains; cheapness by warm-starting is available everywhere, and it is the weaker of the two.

15.4 Worked example: seven pipes, one district

Take the two-district network of Chapter 8: a reservoir at 60 m feeding junctions 2 and 3 through two mains and a link, a transfer main 3–4, and junctions 4, 5 and 6 in a loop, seven pipes in all, with the nominal demands of that chapter summing to 41 kg/s. Each pipe may be out of service over the horizon — a burst with its isolating valves shut, or a planned shutdown — independently with probability 0.02. Figure 15.1 shows the network with its base-case flows and the head margin at each junction above the service head of 30 m.

The consequence of a failure set is the demand the network cannot serve, in kg/s. A junction cut off from the reservoir loses all of its demand. A junction still connected loses part of it if its head falls, by Wagner's pressure-dependent rule: the fraction it can take is $\sqrt{h/h_{\min}}$ between the junctions' common elevation of 0 m and the service head $h_{\min} = 30$ m, one above it and zero below. The heads come from a demand-driven solve at the nominal demands and the rule is applied afterwards, which is what a utility's model does and which overstates the loss: a junction that takes less water also draws the head back up. Every number is from `ch15_outages.py`.

All 128 sets, exactly. Seven pipes give 128 failure sets, and the script solves every one: a connectivity check, a demand-driven solve on the surviving connected part, and the pressure-dependent score. Table 15.2 gives the seven single outages and the eight sets that contribute most to the risk.

The network is not secure against a single outage: four of the seven lose service somewhere, and two of them lose most of it. The loss of the main R–3 is the worst. Nothing is disconnected

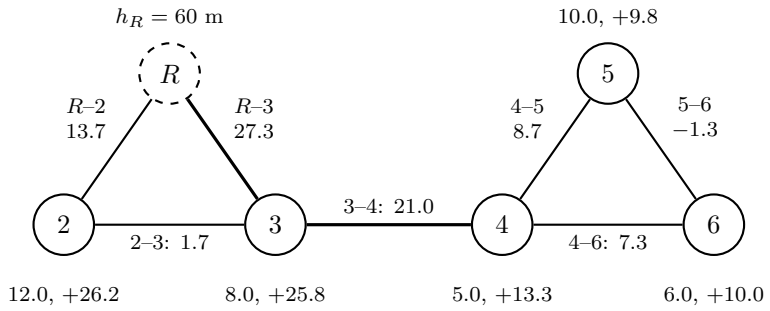


Figure 15.1: The two-district network of Chapter 8. Pipe labels are base-case flows in kg/s; junction labels are the demand in kg/s and the head margin above the 30 m service head. The two thick pipes are the ones whose loss costs most. Every pipe fails independently with probability 0.02. The transfer main is the only path from district B to the reservoir: its loss cuts junctions 4 to 6 off entirely.

by it: every junction is still joined to the reservoir through R–2 and the link 2–3. What fails is pressure. The 150 mm main R–2 passing the whole 41 kg/s loses 28.9 m, leaving 31.1 m at junction 2, and the 100 mm link 2–3 would lose 82.1 m passing the 29 kg/s that must then cross it. There is no head to pay that with. The demand-driven solve puts junction 3 at −51 m and junctions 4 to 6 near −65, and the rule scores their whole 29 kg/s as unserved. That is the first thing this example has to say that a power network would not. Head loss goes as the 1.852 power of flow, so a redistribution that a linear network would absorb with a proportionate overload a water network cannot absorb at all: the margin does not degrade, it runs out. The negative heads are of course not physical — water does not arrive at a negative pressure, it does not arrive — and they are the price of scoring a demand-driven solve after the fact rather than solving with pressure-dependent demands from the start. What they are reliable about is the verdict, not the number.

The loss of the transfer main is next, and it is the other kind of event: nothing is starved, district B is simply cut off, and its 21 kg/s is lost to disconnection rather than to pressure. The loss of the link 2–3 or of the pipe 5–6 costs nothing, because each carries under two kilograms a second and the junctions they serve are reached either way.

Risk and coverage. The risk over all 128 sets is $\mathbb{E}[C] = 1.048$ kg/s, two and a half percent of the demand, and the probability of any consequence at all is 0.078. Figure 15.2 draws the two coverage curves against enumeration depth: the probability retained by the sets of size at most k, and the fraction of the risk they capture. The single outages carry 99.21 percent of the probability and 87.92 percent of the risk; adding the pairs brings the risk to 99.43 percent; the triples bring it to 99.99. The risk curve lags the probability curve because the consequence grows with the size of the set, so the rare large sets carry more risk per unit of probability than the common small ones. On this network the lag is a dozen percent at $k = 1$, and most of that dozen is one pair: both mains out, which disconnects everything and loses all 41 kg/s.

The worst sets, and one that is not. The worst single outage is the main R–3, at 29.0 kg/s. The worst pair is both mains, at 41.0: with neither R–2 nor R–3 in service nothing is joined to the reservoir at all, and the whole demand is lost. Four of the six pairs containing R–3 cost 29.0,

Table 15.2: The two-district network: single outages, and the eight failure sets that contribute most to the risk $\mathbb{E}[C] = 1.048$ kg/s. C is the demand the network cannot serve.

pipe out	unserved [kg/s]	share of demand	$\mathbb{P}(S)\,C$
R–2	0	0%	0
R–3	29.000	71%	0.5138
2–3	0	0%	0
3–4	21.000	51%	0.3721
4–5	1.903	4.6%	0.0337
4–6	0.118	0.3%	0.0021
5–6	0	0%	0
failure set	$\mathbb{P}(S)$	C	**share of risk**
$\{R$–3$\}$	1.8×10^{-2}	29.00	49.0%
$\{3$–4$\}$	1.8×10^{-2}	21.00	35.5%
$\{4$–5$\}$	1.8×10^{-2}	1.90	3.2%
$\{R$–2, R–3$\}$	3.6×10^{-4}	41.00	1.4%
$\{R$–3, 2–3$\}$	3.6×10^{-4}	29.00	1.0%
$\{R$–3, 4–5$\}$	3.6×10^{-4}	29.00	1.0%
$\{R$–3, 4–6$\}$	3.6×10^{-4}	29.00	1.0%
$\{R$–3, 5–6$\}$	3.6×10^{-4}	29.00	1.0%

the worst single outage again, because once R–3 is gone junctions 3 to 6 are already unserved and there is nothing left for a second failure to take.

The sixth is worth stopping on. Lose R–3 *and* the transfer main and the consequence falls to 21.0 kg/s, less than losing R–3 alone. District B is now disconnected, so it draws nothing; only junction 3's 8 kg/s has to cross the link 2–3, which loses 7.6 m doing it rather than 82, and junction 3 is served. The second failure relieved the first. The consequence is therefore *not monotone in the failure set*, and any pruning rule that assumes a superset is at least as bad as its subsets — which is the rule that makes most $N - k$ searches affordable — is wrong on this network. A conserved quantity that stops being drawn when its path is cut has no analogue in a network whose demand is enforced regardless; this is the second thing water says that power does not.

On seven pipes these maxima are found by looking. On seven hundred they are the interdiction problem of §15.7, and the non-monotonicity is what makes that problem harder here than there.

Uncertain demands. So far the demands were their nominal values and each branch's consequence was a number. Give each demand the ±2 kg/s prior of Chapter 8, and each branch's consequence becomes a distribution, which the script samples. The probability of any consequence rises from 0.078 to 0.125, and the risk from 1.048 to 1.177 kg/s. The rise in probability is the base case: with nominal demands the intact network serves everything with almost ten metres of margin at junction 5, but with the demands uncertain it has a 5.6 percent chance of falling short somewhere, worth 0.06 kg/s on average. Demand uncertainty adds risk where the margins are thin, and it adds it to the branch that has all the probability.

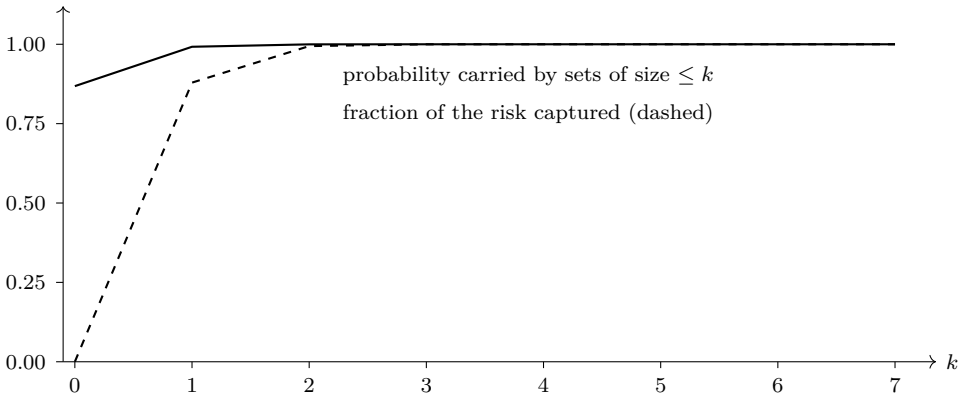


Figure 15.2: Coverage against enumeration depth on the two-district network: the probability carried by the failure sets of size at most k (solid), and the fraction of the total risk they capture (dashed). The risk lags the probability because larger sets have larger consequences.

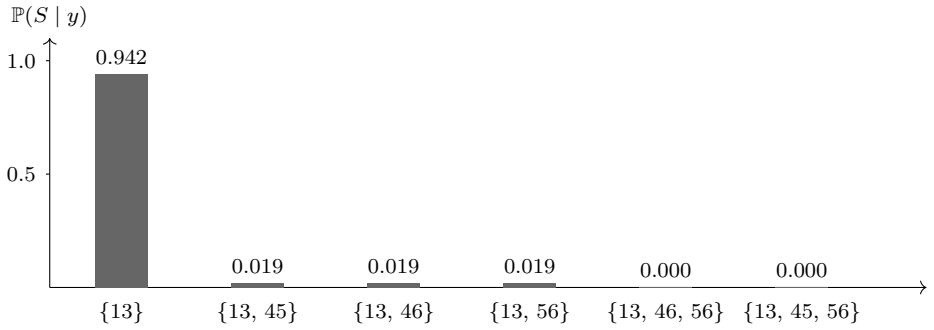


Figure 15.3: The posterior over failure sets on the six-bus network after two meters read $F_{12} = 4.10$ and $F_{34} = 2.10$, each to ± 0.05: the six most probable sets. The outage of line 1–3 is identified at 94 percent; the residual weight is on that outage together with a failure inside region B that the two meters cannot see.

15.5 The posterior over failure sets

Everything so far was prior: the risk before any reading. In operation the meters are reading, and the question changes from "which failures might occur" to "which failures have occurred, and what is the risk now". The hybrid model answers it the way Chapter 7 answered the one-line version: each failure set is a branch, each branch has an evidence given the readings, and the posterior over sets is the prior times the evidence, normalized. A meter on a failed line reads zero; a meter on a surviving line reads that branch's flow; and the posterior weighs every set by how well its flows explain what the meters say.

Let the meters on lines 1–2 and 3–4 read 4.10 and 2.10, each to ± 0.05; these are the flows the network carries with line 1–3 out and nothing else wrong. Figure 15.3 gives the posterior over the 128 sets. The outage of line 1–3 alone takes 94 percent: its prior was 1.8 percent, and two readings raised it by a factor of fifty, because no other single set puts 4.1 on line 1–2 while keeping 2.1 on the tie. The remaining six percent is divided equally among three sets, line 1–3 together with each of the three lines inside region B. The meters cannot distinguish these from

the single outage, and the reason is Chapter 8's: a failure inside region B that islands no bus changes nothing that region B tells the rest of the network, so the tie flow stays at 2.10 and the meters are blind to it. Their posterior weight is their prior weight relative to the single outage, one extra failure at two percent each, and only a meter inside region B would separate them. The posterior says which meter to buy next, as Chapter 11 said it would.

The risk conditional on the readings is $\mathbb{E}[C \mid y] = 3.52$ per unit against a prior risk of 0.165: not because the world has become more dangerous but because the readings have revealed that the largest single consequence has already occurred, and the posterior is nearly certain of it. The probability of a consequence given the readings is one. This is the operational form of the $N - k$ question: not "what could fail" but "what has failed, how sure are we, and what would the next failure cost". The last is a branch over the surviving lines with the posterior as its prior, and on this network the expected additional consequence of one more failure in the next horizon is close to zero, since the worst has happened and several further failures would island load that is already overloading the lines that serve it. That is a fact about this small network, not a rule; on a larger one the next failure after a first is often the one that cascades, and the conditional risk is the number to watch.

Principle 15.1. *Given readings, the failure sets have a posterior: prior times evidence per branch. It identifies what has failed, says what the meters cannot tell apart and why, and turns the prior risk into the conditional risk that operation needs.*

15.6 Enumeration by regions

The six-bus network has 128 failure sets and each was solved on the whole network. It did not have to be. The network is two regions and a tie, and its failure sets are the product of the regions' failure sets: region A's three lines give eight, region B's three lines give eight, the tie gives two, and $8 \times 8 \times 2 = 128$. By Chapter 8, each region's contribution to any branch is its message on the ports of the tie, a two-by-two precision and a two-vector, computed from the region alone. So the 128 branches need eight region solves for A, eight for B, and 128 combinations of two small messages with the tie's factor: sixteen region solves in place of 128 network solves, and the combinations are two-by-two arithmetic.

The saving is a factor of eight on six buses and it grows with the regions. A network of m regions with L_r failing lines in region r has $\prod_r 2^{L_r}$ failure sets, but only $\sum_r 2^{L_r}$ region solves, and the sum is smaller than the product by a factor that is exponential in the number of regions. With pruning by prior weight inside each region, so that only the sets of size up to k_r are kept, the count is $\sum_r \sum_{j \le k_r} \binom{L_r}{j}$ region solves, which is what makes a hundred-line network with three lines out affordable when the lines are spread over ten regions and not when they are treated as one.

The script checks the one thing that makes this exact: that a failure inside region B changes region B's message only through what crosses the tie. With no failure in B, or with line 5–6 out, the tie carries 2.100 and the message is the same. With lines 4–5 and 4–6 both out, buses 5 and 6 are islanded, 1.60 of load is shed, and the tie carries 0.500: the message has changed, and it has changed because region B's total draw has changed, which is what its message carries. Region A's eight solves are reused across all of region B's eight failure sets, and region B's across all of A's, exactly as §8.5 arranged; and the posterior of §15.5 is computed the same way, one evidence per combination of region messages, with the readings inside each region entering that region's message and nowhere else.

15.7 The worst k-set as a query

The third question, $\max_{|S|=k} C(S)$, is an optimization over $\binom{L}{k}$ sets, and it is the one question here that the prior does not help with: the worst set is the worst whatever its probability. On small systems it is found by enumeration, as above. On large ones it is the *interdiction* problem, in which an adversary chooses the k components to disable so as to maximize the damage, and it is hard in general.

Three things the framework offers bear on it. The consequence per set is cheap when it is a rank-one update per component removed, so enumeration reaches further than a re-solve per set would. The convexity of §14.7 gives bounds, and the two directions of bound do different jobs. When C is the value of a linear program in the surviving capacities, the surrogate of Chapter 14 bounds it from below on any set. A lower bound certifies a set already found: its consequence is at least that much. It cannot prune. In a branch-and-bound that maximizes over sets, a branch is a partial set, and it may be discarded only when an upper bound on the consequence of every completion of it falls below the best value found so far. A score that can undercut the true completion value discards branches that may hold the optimum, and a search pruned by it returns a best-found set, not a proved one. And the region structure of Chapter 8 separates the search: a failure inside a region changes that region's message to its ports and nothing else, so the worst set can be sought region by region and then across the ports, at a cost that scales with the regions rather than with the whole.

For the smallest k the global optimum is enumerable, and there the answer is proved. A companion interdiction study in the power domain, committed to the repository that accompanies the book, works a published stochastic benchmark on a 73-bus network with 216 candidate components and 200 background outage scenarios. It proves the $k = 1$ optimum by evaluating all 216 candidates. It proves the $k = 2$ optimum by evaluating all 15,602 candidates that remain once interchangeable components are merged, at 3,040,242 linear programs. For $k = 3$ to 10 a best-first search reproduces the published optima. Its pruning score is not a valid upper bound, so its certificate is conditional on that score. Every one of those runs stopped at its budget of 500,000 evaluated outage states, and no bound on the optimality gap is reported. The same study finds that ranking the components one at a time and taking the top k recovers the published objective at every k, at about 42,600 outage states whatever k is. The network is capacity-adequate, so the damage a set does is governed by how much reserve it removes, and the problem behaves like a knapsack rather than a search over interacting failures. Beyond the enumerable k, the honest form of the answer is a best-found set together with what certifies it; when nothing certifies it, the answer says so.

> **Principle 15.2.** *A failure set is a branch, its consequence is an intervention's readout, and its probability is a product of availabilities. Risk is the mixture, ranking is the product, coverage is the binomial tail, and the worst case is a search that the prior does not shorten but the physics does.*

15.8 Common cause and correlated failure

Independence was the assumption of equation (15.1), and it is the assumption most often wrong. Components share a weather, a supply, a maintenance crew, a design flaw; when the shared thing fails, they fail together, at a rate that the product of their individual rates does not approach. Two models capture this, and both fit the factor graph as a variable with a prior.

15.8.1 A shared shock

Let a common cause $c \in \{0, 1\}$ occur with probability r, and let every component fail when it does, in addition to failing independently with probability q when it does not. This is the simplest Marshall–Olkin shock model: independent shocks, one hitting every component and L hitting one each. The probability that a given pair both fail is

$$\mathbb{P}(a_i = 0,\ a_j = 0) \ = \ r + (1 - r)\, q^2, \tag{15.3}$$

against q^2 under independence, and the probability that all L fail is $r + (1 - r)q^L$ against q^L. For r comparable to q^2 the pairs are twice as likely as independence says; for r comparable to q the shared cause dominates every joint failure, and no amount of redundancy helps, because redundancy protects against independent failures and the shared cause fails everything at once. In the factor graph, c is one discrete variable with a prior $\mathbb{P}(c = 1) = r$, and every availability's prior becomes conditional on it: $\mathbb{P}(a_i = 0 \mid c) = 1$ if $c = 1$ and q otherwise. The branch count doubles, from 2^L to 2^{L+1}, and the shared-cause branch is one of them.

15.8.2 A mixture over the rate

Let the failure rate itself be uncertain, taking the value q_w with probability π_w over a set of conditions w: a hot day, a storm, a normal day. Conditional on the condition the failures are independent; unconditionally they are not. The prior over a set is the mixture

$$\mathbb{P}(S) \ = \ \sum_{w} \pi_w \, q_w^{|S|} (1 - q_w)^{L - |S|}, \tag{15.4}$$

which is exchangeable but not independent. The induced correlation between two components' failures is

$$\rho \ = \ \frac{\mathbb{E}_w[q_w^2] - \bar{q}^{\,2}}{\bar{q}(1 - \bar{q})}, \qquad \bar{q} = \mathbb{E}_w[q_w], \tag{15.5}$$

which is the variance of the rate across conditions divided by the Bernoulli variance at the mean rate.

Proof. Two failures are indicators X_i, X_j with $\mathbb{E}[X_i] = \mathbb{E}_w[\mathbb{E}[X_i \mid w]] = \bar{q}$ and, by conditional independence, $\mathbb{E}[X_i X_j] = \mathbb{E}_w[q_w^2]$. Then $\mathrm{Cov}(X_i, X_j) = \mathbb{E}_w[q_w^2] - \bar{q}^2 = \mathrm{Var}_w[q_w]$, and each indicator's variance is $\bar{q}(1 - \bar{q})$. □

The correlation is small when the rate varies little, and it is not the point. The point is the tail: a condition with a high rate makes large failure sets far more probable than the mean rate suggests, since $q_w^{|S|}$ grows with $|S|$ faster for the larger q_w. A ten-percent chance of a day on which the rate triples multiplies the probability of a triple failure by a factor that the correlation of a few hundredths does not hint at. In the factor graph, w is one discrete variable with prior π_w, and the availabilities' priors are conditional on it, exactly as for the shock; the difference is that the shock fails everything and the condition raises everything's rate.

15.9 Worked example: a booster station

A booster station passes 12 kg/s into a district against a static lift of 15 m, and $n = 4$ identical pumps share the work, each with its own suction and delivery pipework of conductance 1.5 kg/s

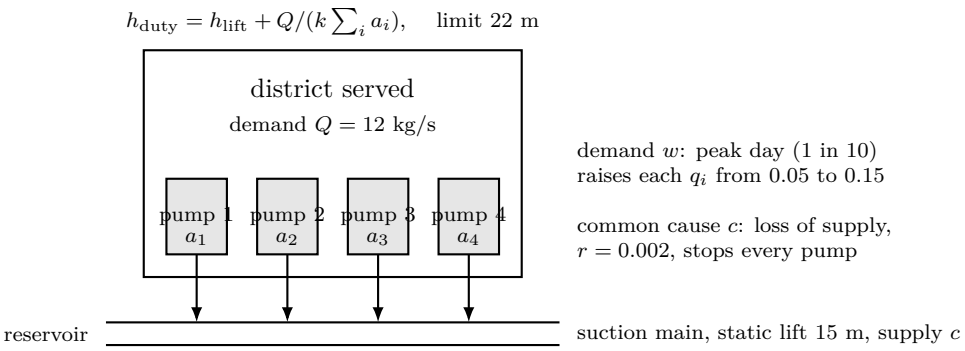


Figure 15.4: A district fed by four identical pumps sharing one suction main and one power supply. Each pump has an availability a_i; the loss of supply c is a common cause; a peak-demand day w raises every pump's failure rate together.

per metre. The head the station must generate has to stay below the 22 m its pumps can deliver. Each pump fails over the horizon with probability 0.05; the station's power supply, which all four share, fails with probability 0.002, and when it does no pump runs. On a peak-demand day, which is one day in ten, the pumps run harder and longer and their failure rate is 0.15. Figure 15.4 draws the station.

The consequence. With m pumps running, the station's pipework is m paths in parallel, a conductance $m\,k$ carrying the whole Q, and the head the station must generate is the closed form

$$h_{\text{duty}}(m) \;=\; h_{\text{lift}} + \frac{Q}{m\,k}, \tag{15.6}$$

which is the pumping chain of Chapter 1 with one node. The duties are 17.0, 17.7, 19.0 and 23.0 m for four, three, two and one pumps, and unbounded for none. Two pumps keep the station inside what it can deliver; the station is $N-2$ tolerant, which is more redundancy than most.

Independent failures. The limit is exceeded when three or more pumps fail, which under independence at $q = 0.05$ has probability $\binom{4}{3}q^3(1-q) + q^4 = 4.8 \times 10^{-4}$: once in two thousand horizons.

With the common cause. The supply fails with probability 0.002, and when it does every pump is down whatever its own state. The probability of exceeding the limit is $r + (1-r) \times 4.8 \times 10^{-4} = 2.5 \times 10^{-3}$, five times the independent figure, and four fifths of it is the supply. The four pumps' redundancy protected against the independent failures to one part in two thousand; against the shared cause it protected against nothing, and the shared cause, at one part in five hundred, is now the risk.

With the demand. On a peak day the exceedance probability is $\binom{4}{3}(0.15)^3(0.85) + (0.15)^4 = 1.2 \times 10^{-2}$, twenty-five times the ordinary-day figure; averaged over the one-in-ten peak days, the unconditional probability is 1.6×10^{-3}, three and a half times the independent figure. The correlation the mixture induces between two pumps' failures, by equation (15.5), is 0.016, a number that would pass unnoticed in any table of correlations. The tail is where it shows: the

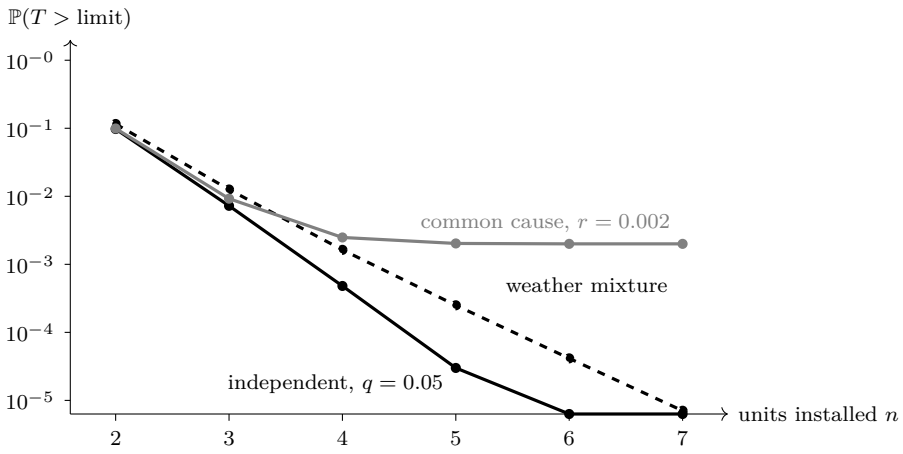


Figure 15.5: Probability of exceeding the station's duty limit against the number of pumps installed, on a logarithmic scale, under three priors: independent failures at $q = 0.05$; the demand mixture; and the loss of supply at $r = 0.002$. Adding pumps drives the independent risk down by a factor of twenty per pump and leaves the common-cause risk at its floor.

probability that three named pumps fail together is $q^3 = 1.25 \times 10^{-4}$ under independence and $0.1 \times 0.15^3 + 0.9 \times 0.05^3 = 4.5 \times 10^{-4}$ under the mixture, nearly four times as much, and it is the triple failures that exceed the limit. The peak day is also the day the district can least afford it, which the arithmetic above does not know and the operator does.

Figure 15.5 draws the exceedance probability against the number of pumps installed, from two to seven, under the three priors. Under independence each added pump divides the risk by about twenty, and by seven pumps it is below one in a million. Under the demand mixture the curve falls too, but more slowly, since the peak-day rate is three times the ordinary one and each added pump divides the peak-day risk by only about six. Under the common cause the curve flattens at 0.002 and stays there: the seventh pump is as useless as the fifth, because the failure that matters takes all of them. An operator who reads only the independent curve buys pumps; one who reads the common-cause curve buys a standby generator.

> **Principle 15.3.** *Redundancy protects against independent failures and against nothing else. A shared cause puts a floor under the risk that no number of components lowers, and a shared condition fattens the tail far beyond what its correlation suggests. Both are one discrete variable in the graph, and the branches say what the redundancy is worth.*

15.10 In practice

Contingency lists. Operators do not enumerate; they maintain a list of the contingencies worth checking, built from experience and from screening. The ranking of §15.4 is the principled form of that list: every single outage costs one rank-one update, the pairs among the top singles cost two, and the list is the sets with the largest $\mathbb{P}(S)C(S)$ to whatever depth the tail requires. What the probabilistic version adds to the traditional list is the weight, so that a likely small event and an unlikely large one are compared on one scale, and the coverage figure, so that the list's completeness is a number rather than a judgment.

Screening by distribution factors. For a large network, the LODF matrix of Proposition 15.2, L columns of one solve each, screens every single outage for every line's post-outage flow at no further cost, and the pairs can be screened by a second application before any is solved exactly. The screen is exact in the DC approximation and a guide in the AC one, where the flows after an outage must be re-solved for the sets the screen flags.

Where to stop. Report the enumeration depth and the tail it leaves. A risk quoted without its unexamined probability is a risk quoted without an error bar. For a hundred components at one percent the depth is three for one-percent accuracy; for five hundred at half a percent no depth is affordable and the honest report is a sampled risk with a sampling error, or a risk over regions with the regions' messages combined exactly.

The common mistakes. Three recur. The first is to treat the $N-1$ verdict as the risk: a system can be $N-1$ secure and carry most of its risk in pairs, or insecure to a single failure that almost never happens. The second is to model failures as independent because the data on their dependence is poor; the shared loop of §15.9 raised the risk fivefold at a probability of one in five hundred, and a modeler who does not know r should carry it as a variable with a wide prior rather than set it to zero. The third is to report a worst case without a bound: on any system too large to enumerate, the worst set found is a lower bound on the worst set that exists, and it should be reported as one.

What to report. For each question, the number and its qualifier: the risk with its unexamined tail; the ranked list with its depth; the worst set with its bound; and, always, the do-nothing baseline of Chapter 13, since every mitigation proposed against the ranked list must be compared with leaving the list as it is.

15.11 What is missing

Every failure in this chapter was instantaneous and every consequence was a steady state. Real failures unfold: a line trips, flows redistribute, another line overloads and trips seconds later, and the cascade's consequence depends on the order and the timing. That needs the dynamics of Chapter 16. And every enumeration here was over a whole system; Chapter 17 organizes the enumeration by the regions of Chapter 8, where a region's failure sets are enumerated inside the region and combined at its ports, and Chapter 18 asks how large a region can be before that stops working. Part IV ends here; the operator's questions have each been asked, and each has been a readout of one posterior, a branch of it, or a search over its branches.

Notes and further reading

Power-system reliability evaluation, with independent-failure priors, contingency enumeration, and the loss-of-load indices that are this chapter's risk and exceedance probability, is treated at book length by Billinton and Allan [8]. The $N-1$ criterion and its contingency lists are standard operating practice; Wood, Wollenberg and Sheblé [101] derive the line outage distribution factors of Proposition 15.2 and their use in contingency screening, and Guler, Gross and Liu [28]

generalize them to multiple simultaneous outages. The reading of the LODF as a Sherman–Morrison update, and of a failure set as a branch of the hybrid model whose consequence is an intervention's readout, is the book's.

The $N-k$ interdiction problem, finding the k components whose loss maximizes the consequence, is studied in a probabilistic form by Sundar, Coffrin, Nagarajan and Bent [89]. Sundar, Mastin, Garcia, Bent and Watson [90] pose its stochastic form on the RTS-GMLC network: the consequence is the minimum load shed of a DC dispatch, averaged over 200 background outage scenarios of weight 1/200 each, for $k = 1$ to 10. A companion study in the power domain, committed to the repository that accompanies the book, replicates that benchmark, with the findings summarized in §15.7. The same study also conditions the scenario distribution on evidence of fuel and storm regimes. Under fuel evidence the worst single outage moves from the generator at bus 121 to the one at bus 118, and no further linear program is solved, because only the scenario weights change. The shock model of §15.8 is Marshall and Olkin's [59]; the mixture over a shared rate is the standard exchangeable model of common-cause failure in reliability engineering, and the correlation formula (15.5) is elementary. That a shared condition shows in the tail and not in the correlation is the book's emphasis, and Figure 15.5 is its evidence.

Summary

- Four questions: risk, ranking, worst case, coverage. The $N-1$ check is the $k=1$ slice of the third.
- Under independent failures the prior decays geometrically in set size; coverage by depth k is the binomial tail. Seven lines are covered by 29 sets; a hundred need 161,700 for 98 percent; five hundred cannot be enumerated.
- A failure is an intervention, its consequence a readout, and in a DC network a rank-one update: the LODF, derived and checked to 10^{-15}.
- Six buses, seven lines, 128 sets: risk 0.165, singles carry 90 percent of it, pairs bring it to 99.6; the worst single is the heaviest line, the worst pair adds a region's overload. Load uncertainty doubles the probability of any consequence by threatening the base case.
- The worst k-set is a search the prior does not shorten; the rank-one consequence, the convex bound and the region structure do.
- A shared cause puts a floor under the risk that redundancy cannot lower: four pumps protect the district to one in two thousand against independent failures and to one in five hundred, five times worse, against a loss of supply at one in five hundred. A shared condition fattens the tail fourfold at a correlation of 0.016.

Exercises

Exercise 15.1. Derive equation (15.2) directly from the Sherman–Morrison identity applied to $\mathbf{B}'' = \mathbf{B}' - B_o \mathbf{a}_o \mathbf{a}_o^{\top}$, and show that the transfer argument in the proof of Proposition 15.2 and the algebraic derivation give the same factor. Identify $\mathrm{PTDF}_{o,o}$ in terms of $\mathbf{a}_o^{\top} \mathbf{B}'^{-1} \mathbf{a}_o$.

Exercise 15.2. For L components at equal rate q, bound the binomial tail $\mathbb{P}(|S| > k)$ from above by a geometric series and find the smallest k for which the bound is below 10^{-6} when $L = 100$, $q = 0.01$. Compare with the exact tail in Table 15.1.

Exercise 15.3. On the two-district network, add a second transfer main from junction 6 to the reservoir, of the same diameter as the existing one, and recompute the 256 failure sets. Which single outage is now the worst, does the loss of the first transfer main still cut district B off, and how does the risk change? Explain in terms of the region message of Chapter 8.

Exercise 15.4. Give the booster station a second power supply feeding two pumps each, with each supply failing at 0.002. Compute the exceedance probability under the two-loop common-cause model and compare with the single loop and with independence. At what per-supply failure rate does a second supply stop being worth more than a fifth unit?

Exercise 15.5. For the weather mixture with conditions w and rates q_w, show that the probability of a set of size k under the mixture, relative to the same set under the mean rate $\bar{q}$, is at least one, with equality only when the rate does not vary, and find the ratio for $k = 3$ in the room example. (Hint: Jensen's inequality applied to $q \mapsto q^k$.)

Solutions to selected exercises

Exercise 15.1. Sherman–Morrison gives $(\mathbf{B}' - B_o \mathbf{a}_o \mathbf{a}_o^{\mathsf{T}})^{-1} = \mathbf{B}'^{-1} + \dfrac{B_o \, \mathbf{B}'^{-1}\mathbf{a}_o \mathbf{a}_o^{\mathsf{T}} \mathbf{B}'^{-1}}{1 - B_o \, \mathbf{a}_o^{\mathsf{T}} \mathbf{B}'^{-1} \mathbf{a}_o}$. The post-outage angles are $\boldsymbol{\theta}' = \mathbf{B}''^{-1}\mathbf{P} = \boldsymbol{\theta} + \dfrac{B_o \, \mathbf{B}'^{-1}\mathbf{a}_o \, (\mathbf{a}_o^{\mathsf{T}}\boldsymbol{\theta})}{1 - B_o \mathbf{a}_o^{\mathsf{T}} \mathbf{B}'^{-1} \mathbf{a}_o}$. Now $\mathbf{a}_o^{\mathsf{T}}\boldsymbol{\theta}$ is the pre-outage angle difference across line o, so $B_o \mathbf{a}_o^{\mathsf{T}}\boldsymbol{\theta} = F_o$; and $\mathbf{B}'^{-1}\mathbf{a}_o$ is the angle response to a unit transfer along o, so $B_e \mathbf{a}_e^{\mathsf{T}} \mathbf{B}'^{-1} \mathbf{a}_o = \mathrm{PTDF}_{e,o}$ for every line e, including $e = o$. The flow on line e after the outage is $B_e \mathbf{a}_e^{\mathsf{T}} \boldsymbol{\theta}' = F_e + \dfrac{\mathrm{PTDF}_{e,o} \, F_o}{1 - \mathrm{PTDF}_{o,o}}$, which is equation (15.2); the transfer argument's t is $F_o / (1 - \mathrm{PTDF}_{o,o})$, the same scalar. In particular $\mathrm{PTDF}_{o,o} = B_o \mathbf{a}_o^{\mathsf{T}} \mathbf{B}'^{-1} \mathbf{a}_o$, the fraction of a transfer along o's ends that flows on o itself, which is one exactly when o is the only path.

Exercise 15.2. Each binomial term satisfies $\binom{L}{j+1} q^{j+1} (1-q)^{L-j-1} = \binom{L}{j} q^j (1-q)^{L-j} \cdot \dfrac{(L-j)\,q}{(j+1)(1-q)}$, and for $j \geq k$ the ratio is at most $\rho_k = \dfrac{(L-k)\,q}{(k+1)(1-q)}$, which is below one once $k + 1 > (L-k)q/(1-q)$. Then the tail is bounded by the geometric series $\mathbb{P}(|S| > k) \leq \mathbb{P}(|S| = k) \, \dfrac{\rho_k}{1 - \rho_k}$. For $L = 100$, $q = 0.01$: at $k = 5$, $\mathbb{P}(|S| = 5) = 2.9 \times 10^{-3}$ and $\rho_5 = 95 \times 0.01 / (6 \times 0.99) = 0.160$, giving a bound of 5.5×10^{-4} against the exact 5.35×10^{-4}; at $k = 8$, $\mathbb{P}(|S| = 8) = 7.4 \times 10^{-6}$, $\rho_8 = 0.103$, bound 8.5×10^{-7}, below 10^{-6}. So $k = 8$ suffices, and the bound is within ten percent of the exact tail at every depth because the ratio is small. The $\binom{100}{8} = 1.9 \times 10^{11}$ sets of size eight are the reason enumeration to that depth is not done; the bound says how much is lost by stopping earlier.

Part V

Time, hierarchy, and scale

Chapter 16

Dynamics and temporal factors

Every balance in the book so far had its accumulation term set to zero. The junction did not store water; the reservoir did not draw down; the state was whatever the flows made it, instantly. Real systems store, and what they store changes the questions. A step in the source does not move the discharge's head at once but over minutes; a district cut off from its supply does not run dry at once but when its reservoirs have drained; a reading taken now is evidence about the state a minute ago and a prediction about the state a minute hence. This chapter makes the accumulation term live. It shows that integrating the resulting equations in time is a sequence of the steady solves the book already has; that the graph of a system over many time steps is the graph of one step, copied and chained, so that everything Part III said about separators applies with the stored state as the separator between past and future; that filtering, smoothing and prediction are eliminations on that chain, and the Kalman filter is the rank-one update of Chapter 11 preceded by one Schur complement; and that a cascade, the failure that Chapter 15 could not model, is a sequence of events located on a trajectory. Two worked examples carry it: the pumping chain with a vessel on each node, filtered and smoothed through a drifting source, and a water district after the loss of its transfer main, whose two service reservoirs drain until one of them shuts.

16.1 The accumulation term made live

Return to the balance law of Chapter 1,

$$\frac{\mathrm{d}X_{n,d}}{\mathrm{d}t} \;=\; \sum_{e \in \mathcal{E}(n)} A_{n,e}\, P_{d,e} + G_{n,d} - L_{n,d}, \tag{16.1}$$

and keep the left side. A node that can accumulate has an extensive state X, and a *storage relation* that ties it to the node's intensive state: $M = \rho A h$ for a vessel of plan area A holding water to a depth h, $E = C\,T$ for a thermal mass of capacity C, $q = C_{\mathrm{el}} V$ for a capacitor. The unknowns now include the storage states, the rows now include the storage relations, and the balance rows at storing nodes are differential equations while every closure and every balance at a zero-storage node is algebraic. The whole is a *differential-algebraic* system,

$$\mathbf{F}(\mathbf{z},\ \dot{\mathbf{z}},\ t) = \mathbf{0}, \tag{16.2}$$

of index one: the algebraic rows can be solved for the algebraic variables given the differential ones, which is the well-posedness of §1.5 applied at each instant. Nothing about the graph has

changed. The storage relation is one more closure, reading X and h; the balance row reads $\dot{X}$ as well as the flows; the factor graph of Chapter 2 gains one variable and one square per storing node.

The chain with masses. Give the discharge of the flow chain a storage capacity $C_1 = 200$ kilograms per metre and the junction $C_2 = 5$. The two balances become

$$C_1\,\dot{h}_1 = G - k\,(h_1 - h_2), \qquad C_2\,\dot{h}_2 = k\,(h_1 - h_2) - k_2\,(h_2 - h_a), \tag{16.3}$$

with the closures substituted in. This is a linear system $\dot{\mathbf{h}} = \mathbf{A}\mathbf{h} + \mathbf{b}$, and its eigenvalues say how it moves: from the script `ch16_dynamics.py` in the book's `scripts` directory, which supplies every number in the chapter, they are -0.0071 and -1.42 per second, time constants of 142 seconds and 0.7 seconds. The discharge responds over minutes; the junction, light and well coupled, follows the discharge within a second. The ratio of the two, about 200, is the system's *stiffness*, and it is the number that decides how the system may be integrated.

16.2 Integration as a sequence of steady solves

To advance the state from time t_n to $t_{n+1} = t_n + \Delta t$, replace the derivative in the differential rows by a difference. The simplest replacement that works for stiff systems is the backward difference,

$$\text{BDF1:} \qquad X^{n+1} - X^n - \Delta t\,\mathcal{B}(\mathbf{z}^{n+1}) = 0, \tag{16.4}$$

where $\mathcal{B}$ is the right side of equation (16.1) evaluated at the new time. The second-order version uses two past values,

$$\text{BDF2:} \qquad 3X^{n+1} - 4X^n + X^{n-1} - 2\Delta t\,\mathcal{B}(\mathbf{z}^{n+1}) = 0. \tag{16.5}$$

Both are *implicit*: the unknown state $\mathbf{z}^{n+1}$ appears inside $\mathcal{B}$, and each step is a nonlinear system in $\mathbf{z}^{n+1}$.

Proposition 16.1 (A time step is a steady solve)**.** *With the storage rows rewritten as equation* (16.4) *or* (16.5) *and every algebraic row unchanged, the system to be solved at each step has the same variables, the same sparsity, and the same Jacobian pattern as the steady-state system of Chapter 1, differing only in the storage rows' entries. Simulation is a loop over the steady solver.*

Proof. The steady system has the rows: balances with $\dot{X} = 0$, closures, storage relations, boundary conditions. Equation (16.4) is the balance row with $\dot{X}$ replaced by $(X^{n+1} - X^n)/\Delta t$, which adds the known X^n to the row's constant and the coefficient $1/\Delta t$ to the row's entry on X^{n+1}; it reads the same variables. Every other row is unchanged. So the Jacobian has the steady Jacobian's pattern with the storage-state columns of the balance rows carrying an extra $1/\Delta t$ (or $3/(2\Delta t)$ for BDF2), and Newton's method with the same assembly and factorization applies. □

The proposition is worth more than its proof suggests. It means there is one model, one assembly and one derivative path for the steady solve, the transient, and, by Chapter 6's Proposition 6.2, the probabilistic estimate at each time. It also means that everything said about the steady Jacobian's sparsity and factorization carries to each step, and, as §16.3 shows, to all steps at once.

Table 16.1: Integrating the chain's step response over 600 s: maximum error against the exact solution. Explicit Euler is accurate below its stability limit of 1.41 s and diverges above it; BDF1 is stable at every step with error proportional to Δt; BDF2's error is proportional to Δt^2 until the junction's fast transient limits it.

method	Δt [s]	max error in h_1 [K]	max error in h_2 [K]
explicit Euler	0.5	0.0091	0.010
	1	0.018	0.033
	2	10^{76}	10^{77}
BDF1	1	0.018	0.013
	5	0.090	0.064
	20	0.34	0.25
BDF2	1	0.0005	0.0083
	5	0.011	0.0091
	20	0.13	0.093

Why implicit. The alternative, the forward difference $X^{n+1} = X^n + \Delta t\, \mathcal{B}(\mathbf{z}^n)$, is explicit: the new state is computed from the old one without solving anything. It is cheaper per step and it is unusable on a stiff system. For a linear system $\dot{\mathbf{h}} = \mathbf{A}\mathbf{h}$ the explicit step multiplies each eigencomponent by $1 + \lambda\Delta t$, whose magnitude exceeds one, so that the component grows without bound, whenever $\Delta t > 2/|\lambda|$; the fastest eigenvalue sets the limit, and on the chain it is $\Delta t < 1.41$ seconds, dictated by a junction whose transient is over in a second and of no interest. BDF1 multiplies by $1/(1 - \lambda\Delta t)$, whose magnitude is below one for every negative λ and every Δt: it is stable at any step, and the step can be chosen for the slow dynamics one cares about. Table 16.1 shows all three on the chain's response to a step in the source from 100 to 120 kg/s, against the exact solution by matrix exponential.

At a twenty-second step, forty times the junction's time constant, BDF1 is within a third of a metre over the whole transient and BDF2 within a tenth. The explicit method at a two-second step has produced numbers of order 10^{76}. The stiff component that the implicit methods damp is the same one the explicit method amplifies, and the book's systems, with light and heavy masses coupled, are stiff as a rule.

Choosing the step. A fixed step is rarely right for a whole transient: fine when the state moves fast, coarse when it settles. The standard control accepts a step when a weighted measure of the change it produced is below one and chooses the next step from the ratio, with a safety factor and a cap on growth. The details are in the notes; what matters here is that the control acts on the same step-change measure for every variable of the graph, and that it shrinks the step wherever a fast trajectory appears, which on a stiff system is exactly where an explicit method would have failed.

16.3 The unrolled graph

Take the factor graph of one step and copy it once per step. The copies are joined by the storage rows, equation (16.4), each of which reads X at two times; every other row reads one time only. The result is the *unrolled* factor graph of the transient, Figure 16.1, and it has a structure that

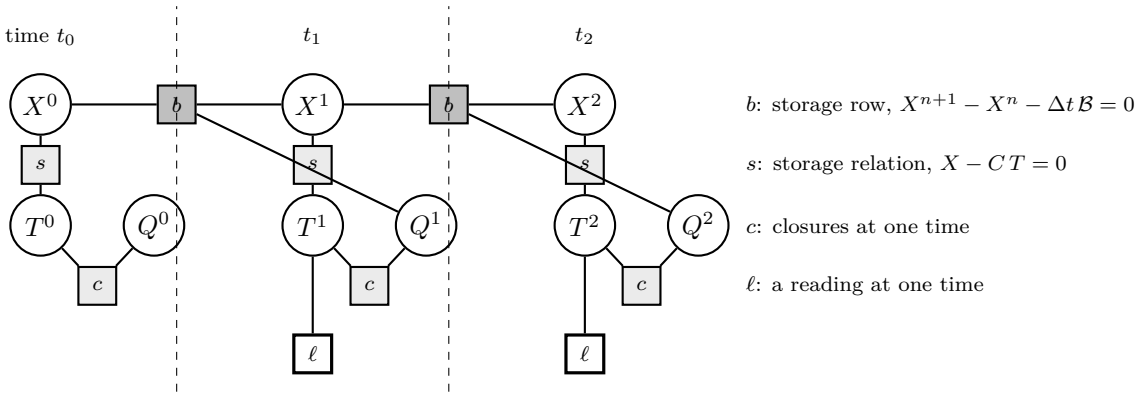


Figure 16.1: The unrolled factor graph of a storing node over three time steps, schematically. Within a step, the storage relation and the closures read variables of that step only. Only the storage rows (dark) read two times, and they read the storage state at both. The dashed cuts pass through the storage states, which are the separators between consecutive steps.

decides everything that follows.

Proposition 16.2 (The stored state separates past from future)**.** *In the unrolled factor graph, the storage states X^n at one time form a separator between all variables at earlier times and all variables at later times. Hence, given X^n, the past and the future are conditionally independent.*

Proof. Every row of the model reads variables of one time step, except the storage rows (16.4), which read X^n, X^{n+1} and the flows and sources at t_{n+1} that enter $\mathcal{B}(\mathbf{z}^{n+1})$. So in the Markov graph, every edge joins two variables of the same step or joins X^n to a variable of step $n+1$. A path from step $m < n$ to step $p > n$ must pass through each intermediate step, and it can enter step n only through X^{n-1}'s edges to step n's variables and leave it only through X^n's edges to step $n+1$. Removing X^n removes every edge out of step n into the future, so no path remains, which is separation, and Proposition 5.1 gives the independence. For BDF2 the storage row reads X^{n-1} as well, and the separator is the pair (X^{n-1}, X^n). □

This is the Markov property of a dynamical system, which filtering theory takes as an axiom; here it is a consequence of the row structure, and it says which variables are the "state" in the sense that matters: the storage states, and nothing else. The heads, flows and sources at time n are algebraic functions of X^n and the inputs and need not be carried.

The precision is block-tridiagonal. With every row Gaussian, the unrolled model's precision has the block structure the proposition implies: a diagonal block per step, and off-diagonal blocks only between consecutive steps, from the storage rows. Ordered by time it is banded, with bandwidth equal to the number of variables in two steps. On the chain with a three-variable state over 180 steps the precision is 543×543 with bandwidth 5 and 1.4 percent of its entries nonzero, and its factorization costs what a chain's does: linear in the number of steps.

16.4 Process noise

The storage row (16.4) is a physics row and, like every physics row in Part II, it is given a width. Its width has a name in the filtering literature, *process noise*, and it means what the closure

width of Chapter 4 meant: the amount by which the model's account of how the state moves may be wrong over one step. Unmodeled disturbances, neglected dynamics, a discretization error, an input that drifts: each is a reason the state at t_{n+1} is not exactly what the row predicts from t_n, and the width is their combined size.

Process noise has a second use, and it is the one the worked example needs. An input that is not constant, a source that drifts, a load that ramps, a parameter that ages, can be modeled as a variable at every time step joined to itself at the next by a storage row of its own, $u^{n+1} - u^n = 0$, with a width: a *random walk*. The width says how far the input may move in one step; the prior on u^0 says where it starts; and the readings, through the physics, then track it. The model does not know the input's trajectory. It knows that the trajectory is continuous at a stated rate, and that is often all that is known.

16.5 Filtering, smoothing, and prediction as eliminations

On the unrolled graph, the three questions of estimation in time are three elimination orders.

Filtering asks for the state at time n given readings up to time n. Eliminate the past forward: fold step 0 into step 1, then the result into step 2, and so on, conditioning on each step's readings as it is reached. By Proposition 16.2 what is passed from step to step is a message on the storage state alone, and for a Gaussian model the message is a mean and a covariance.

Smoothing asks for the state at time n given readings up to a later time N. Eliminate forward to N as in filtering, then back-substitute: the same two sweeps as Chapter 8's tree messages, on a tree that is a chain. Equivalently, solve the whole unrolled precision at once; its banded factorization is the forward sweep and its solve is the backward one.

Prediction asks for the state at a time beyond the last reading. It is filtering with no likelihood factors after the last one: the forward messages continue, growing wider by the process noise at each step, and no reading narrows them.

Proposition 16.3 (The Kalman filter is a Schur complement and a rank-one update)**.** *Let the state at time n have posterior $\mathcal{N}(\boldsymbol{\mu}_n, \boldsymbol{\Sigma}_n)$ given readings to time n, let the storage rows be linear, $\mathbf{x}^{n+1} = \mathbf{F}\mathbf{x}^n + \mathbf{c} + \mathbf{w}$ with $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q})$, and let a reading $y = \mathbf{h}^\mathsf{T}\mathbf{x}^{n+1} + \nu$ arrive. Then the prediction is*

$$\boldsymbol{\mu}_{n+1|n} = \mathbf{F}\boldsymbol{\mu}_n + \mathbf{c}, \qquad \boldsymbol{\Sigma}_{n+1|n} = \mathbf{F}\boldsymbol{\Sigma}_n\mathbf{F}^\mathsf{T} + \mathbf{Q}, \tag{16.6}$$

and the update is the rank-one conditioning of Proposition 11.1 applied to $(\boldsymbol{\mu}_{n+1|n}, \boldsymbol{\Sigma}_{n+1|n})$.

Proof. The joint of $\mathbf{x}^n$ and $\mathbf{x}^{n+1}$ before the reading is the product of the step-n posterior and the storage factor $\exp(-\frac{1}{2}(\mathbf{x}^{n+1} - \mathbf{F}\mathbf{x}^n - \mathbf{c})^\mathsf{T}\mathbf{Q}^{-1}(\cdot))$. Its precision, in the blocks $(\mathbf{x}^n, \mathbf{x}^{n+1})$, is

$$\begin{pmatrix} \boldsymbol{\Sigma}_n^{-1} + \mathbf{F}^\mathsf{T}\mathbf{Q}^{-1}\mathbf{F} & -\mathbf{F}^\mathsf{T}\mathbf{Q}^{-1} \\ -\mathbf{Q}^{-1}\mathbf{F} & \mathbf{Q}^{-1} \end{pmatrix}. \tag{16.7}$$

Eliminating $\mathbf{x}^n$ is the Schur complement of the first block, equation (6.8): the precision on $\mathbf{x}^{n+1}$ becomes $\mathbf{Q}^{-1} - \mathbf{Q}^{-1}\mathbf{F}(\boldsymbol{\Sigma}_n^{-1} + \mathbf{F}^\mathsf{T}\mathbf{Q}^{-1}\mathbf{F})^{-1}\mathbf{F}^\mathsf{T}\mathbf{Q}^{-1}$, which by the Woodbury identity is $(\mathbf{Q} + \mathbf{F}\boldsymbol{\Sigma}_n\mathbf{F}^\mathsf{T})^{-1}$; inverting gives $\boldsymbol{\Sigma}_{n+1|n}$, and the information vector gives the mean. The reading is then one linear row on $\mathbf{x}^{n+1}$, and Proposition 11.1 conditions on it. □

The filter is thus nothing the book had not already built: a region message from the past onto the state, then a reading folded in. The smoother's backward pass is Chapter 8's equation (8.6),

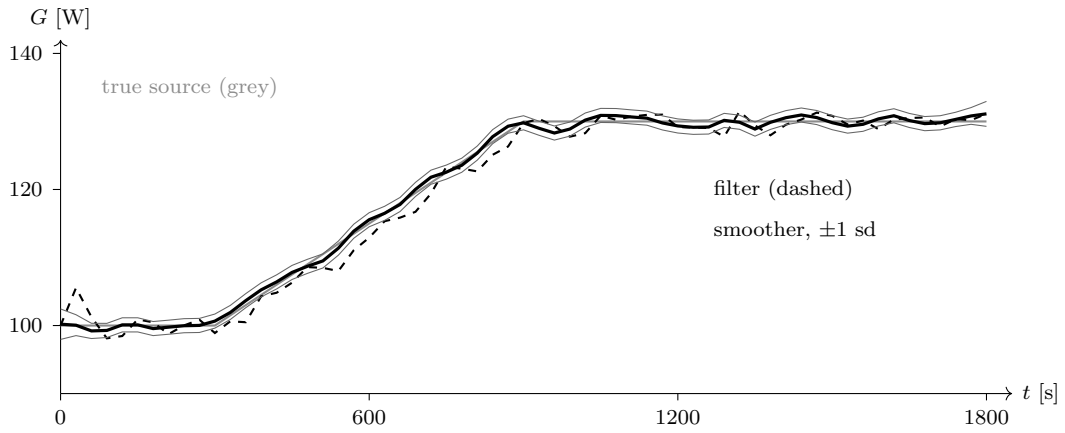


Figure 16.2: The source: true trajectory (grey), the filter's estimate (dashed), and the smoother's estimate with its one-standard-deviation band (black). The filter lags the ramp and catches up when it ends; the smoother, seeing the later readings, follows the ramp.

region by region along the chain. The literature has names for each, Kalman for the forward pass and Rauch, Tung and Striebel for the backward, and they are the sum-product messages of Chapter 8 on a chain of storage states.

Principle 16.1. *Time unrolls the graph into a chain whose separators are the stored states. Filtering is forward elimination, smoothing is forward then backward, prediction is forward without readings, and the Kalman filter is one Schur complement and one rank-one update per step.*

16.6 Worked example: tracking a drifting source

Take the chain with its masses, a source that ramps from 100 to 130 kg/s between 300 and 900 seconds and then holds, and a gauge on the junction reading every 10 seconds with accuracy 0.3 metres. The model does not know the ramp. It carries the source as a random walk with a width of one kg/s per step and starts it at 100 ± 20. The state is (h_1, h_2, G); the transition is the BDF1 step with the source held at its new value across the step; the horizon is 1800 seconds, 180 steps.

Filter against smoother. Figure 16.2 draws the source's true trajectory against the filter's estimate and the smoother's. Table 16.2 gives the errors. The filter tracks the source to 1.5 kg/s root-mean-square and the discharge head to 0.24 metres, from a gauge on the junction that never reads either. The smoother halves both errors, to 0.65 kg/s and 0.13 metres, and halves the reported spread of the source mid-window, from 1.83 to 1.04 kg/s: it has the future readings, which the filter did not, and the future of a slowly moving source is evidence about its present. The filter lags the ramp, as a random-walk prior must when the truth moves in one direction; its standardized innovations, which should average zero, average $+0.51$ during the ramp and $+0.02$ after it. That drift in the innovations is the test of Chapter 12 running in time, and it is how a filter says that its model of the input's motion is too slow.

Table 16.2: Tracking the drifting source: errors of the filter and the smoother over the settled window, and the reported spread of the source at mid-window. The batch solve of the unrolled precision agrees with the smoother to 10^{-9}.

	filter	smoother
rms error of G [W]	1.49	0.65
rms error of h_1 [K]	0.24	0.13
reported sd of G at 900 s [W]	1.83	1.04

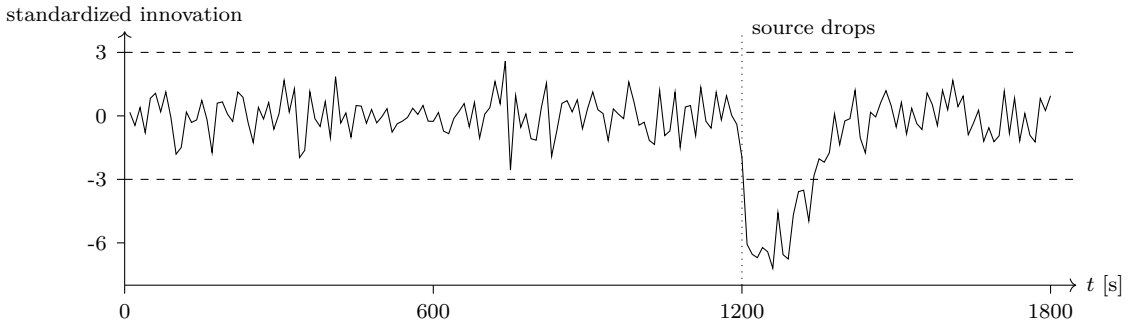


Figure 16.3: Standardized innovations of the filter on the chain when the source drops from 100 to 75 kg/s at 1200 seconds and the model carries the source as a slow random walk. Before the fault the sequence is white within ±3; the first reading after it is six spreads below its prediction, and the sequence stays there until the filter has moved the source.

The batch check. The script also solves the whole unrolled precision, 543 variables, banded, in one factorization, and compares the result with the smoother's backward pass. They agree to 10^{-9}. The smoother and the batch posterior are the same object, computed by the same elimination in a different bookkeeping, which is what Proposition 16.3 and Chapter 8 said.

16.7 Detecting a change in time

The innovation of Chapter 11 was a single number per reading. In time it is a sequence, and the sequence is the most useful diagnostic a running model has. Under an adequate model the standardized innovations are independent standard Gaussians: mean zero, spread one, no memory. Anything the model does not know shows up as a departure from that, and the departure has a shape that says what kind of thing it was. A ramp the model is too slow to follow gives a sustained bias, as §16.6 showed; a bad reading gives one outlier; a sudden change in the system gives a jump that persists until the model has caught up.

Run the chain's filter again with the source held at 100 kg/s and its random-walk width lowered to 0.3 kg/s per step, and at 1200 seconds let the source drop abruptly to 75: a partial failure of the device. Figure 16.3 draws the standardized innovations. Before the fault they have mean 0.00, spread 0.93, and never exceed 2.6 in magnitude over 117 readings. The first reading after the drop, ten seconds later, is 6.1 spreads below what the filter predicted; the next five are 6.5, 6.7, 6.2, 6.4 and 7.2. The alarm fires on the first reading after the fault. The filter, whose prior on the source's motion is slow, takes 130 seconds to bring its estimate within three kg/s

of the new level, and its estimate is 81 kg/s at 1300 seconds and 74.5 at 1500; the innovations return to white as it arrives. The lag is the price of the narrow random walk, and it is the trade the operator makes: a wide walk tracks changes quickly and calls noise a change, a narrow one filters noise and calls a change an anomaly for as long as it takes to believe it. The alarm is right either way; what the width sets is whether the estimate follows the alarm at once or after a decent interval.

Two things distinguish this from a threshold on the gauge. The threshold would fire when the junction had cooled by some fixed amount, minutes after the drop, since the junction follows the discharge's slow transient; the innovation fires on the first reading, because the filter predicted where the junction should be and the junction was not there. And the threshold says the junction is cold; the innovation, through the model, says the source has fallen, which is the thing that failed.

Principle 16.2. *A running model's innovations are its diagnostic in time: white when the model is adequate, biased when it is too slow, spiked when something changed. The alarm fires when the prediction fails, not when the state has drifted far enough to notice.*

16.8 Two hypotheses the steady state cannot separate

The alarm says something changed. Chapter 12 said what to do next: name the candidates, give each a latent variable with a zero-centered prior, and let the evidence choose. Two candidates explain a sudden fall in the junction's readings. The source may have dropped, by an amount a; or the gauge's bias may have jumped, by an amount b. Each is the running model with one more state, born at the step of the alarm with a prior of thirty units, kg/s or metres, on its amplitude: under the first, the jump enters the storage row of the source exactly as a process-noise step does; under the second, it enters the likelihood row of the gauge as an offset.

In steady state the two are the same hypothesis. A source drop of 25 kg/s lowers the junction's steady reading by 12.3 metres, and a gauge bias of -12.3 metres lowers it by the same, and no number of steady readings tells them apart. Chapter 12 met this in Table 12.4, where a leak and a sensor bias were separated only by their priors. In time they are different hypotheses. A bias moves the reading at once and holds it; a source drop moves the discharge over its time constant of 142 seconds, and the junction follows the discharge. The shapes differ, and the evidence weighs shapes.

The evidence of a model in time factors along the chain. Write the joint density of the readings as a product of conditionals, $p(y_1, \dots, y_N) = \prod_n p(y_n \mid y_1, \dots, y_{n-1})$; each factor is the reading's predictive density under the filter, which for the linear-Gaussian model is $\mathcal{N}(e_n;\, 0,\, s_n)$ with e_n the innovation and s_n its variance. Hence

$$\log Z \;=\; \sum_{n=1}^{N} \log \mathcal{N}(e_n;\, 0,\, s_n) \;=\; -\tfrac{1}{2} \sum_{n} \Big(\log 2\pi s_n + \frac{e_n^2}{s_n} \Big), \tag{16.8}$$

which is equation (12.1) computed one reading at a time. The Occam factor is there too: a hypothesis whose extra state is wide inflates s_n at the reading where the state is born, and pays $\frac{1}{2} \log s_n$ for the trunk main it gave itself. The log Bayes factor between two hypotheses is the difference of two such sums, and it can be watched growing reading by reading.

Table 16.3 gives the result for both truths. When the source has dropped, the first reading barely separates the hypotheses: a fall of one reading is a fall, and either explains it, the source

Table 16.3: Log Bayes factor for the true hypothesis against the other, accumulated over the k readings after the alarm, for a source drop of 25 kg/s and for a gauge bias of the same steady effect, -12.3 metres. Amplitude priors of thirty units on both.

k readings after the alarm	1	2	3	5	10	20	30	60
truth: source drop, log (source : bias)	1.3	0.5	0.6	5.8	72	128	129	129
truth: bias jump, log (bias : source)	73	189	314	568	1055	1386	1404	1404

hypothesis ahead by a single nat for having predicted a smaller one. Over the next readings the junction keeps falling, as a body cooling through its time constant does and as a stuck bias does not, and the evidence accumulates at eight nats a reading: seventy-two after ten readings, 128 after twenty. Then it stops. By the thirtieth reading the discharge has reached its new steady state, both hypotheses predict the same reading, and the readings from the thirty-first to the sixtieth add less than a hundredth of a nat between them. Ninety percent of the evidence arrives in the first 150 seconds, which is the time constant. The source hypothesis ends with a jump estimate of -24.5 ± 1.2 kg/s and a source of 74.5; the bias hypothesis, to fit the falling readings, has had to move its random-walk source as well, and ends with a bias of -2.7 metres and a source of 79.9, a compromise that fits the steady state and lost the transient.

When the truth is a bias, the asymmetry shows. A bias of twelve metres moves the reading by forty spreads at once, and a source drop cannot do that: through the discharge's storage, a step of the source moves the junction by three hundredths of a metres per kg/s in the first ten seconds. The first reading alone is seventy nats against the source, and the transient it fails to produce adds the rest.

Two lessons. The distinction between a fault in the physics and a fault in the sensing, which Chapter 12 had to buy with a prior, is bought here with time: the storage rows give each hypothesis a shape, and the shapes are not the same. And the window in which they differ is the system's own time constant. An operator who waits for the new steady state before asking which hypothesis holds has waited past the only readings that could answer.

Principle 16.3. *Hypotheses that the steady state cannot separate are separated by the transient, because storage gives each a shape in time. The evidence factors along the chain, one innovation density per reading, and it accumulates only while the shapes differ: for as long as the time constant, and no longer.*

16.9 Events and cascades

A transient rarely runs to its end undisturbed. A head crosses a limit and a breaker opens; a tank empties and a pump stops; a valve saturates and a regime changes. Each is an *event*: a scalar indicator $\psi(\mathbf{z}, t)$ crosses zero, and at the crossing the model changes, by an intervention in the sense of Chapter 13, a closure replaced or a component removed. Locating the crossing is part of the integration: after each accepted step the indicator is checked for a sign change, the crossing is found by bisection on the interpolated state, and the handler applies the change and the integration continues on the new model.

A *cascade* is a sequence of events each caused by the last. A line trips; the flows redistribute; another line, now overloaded, feeds; it trips in its turn. Chapter 15 treated failure sets as

instantaneous and their consequences as steady states, and could not say which set a cascade would produce or how long it would take. With dynamics it can. The failure set grows one event at a time along the trajectory, and the operator's question changes from "what could fail" to "how long do I have".

16.10 Worked example: two service reservoirs

District B of the two-district network has a service reservoir on junction 5 and a second on junction 6, each floating on the network at a minimum operating level of 40 m of head. The first has a plan area of 140 m^2 and three metres of usable depth, 420 cubic metres; the second has 300 m^2 and three and a half metres, 1050. In normal operation the transfer main supplies the district's 21 kg/s and the reservoirs sit full.

A reservoir is where the accumulation term of §16.1 is not zero. Its balance is

$$\rho A \frac{\mathrm{d}L}{\mathrm{d}t} \;=\; -\,m_{\mathrm{draw}}, \tag{16.9}$$

with A the plan area, L the level above the minimum and m_{draw} the net mass flow the district takes from it. Every number is from `ch16_cascade.py`.

Proposition 16.4 (Time to the minimum level)**.** *If the net draw on a reservoir is constant at $m_{draw} > 0$ from a level L_0 above its minimum, the level falls linearly and the reservoir reaches its minimum operating level at*

$$t_{empty} \;=\; \frac{\rho A L_0}{m_{draw}}, \tag{16.10}$$

which is the usable mass divided by the rate at which it leaves.

Proof. Equation (16.9) with m_{draw} constant integrates to $L(t) = L_0 - m_{\mathrm{draw}} t/(\rho A)$; set $L = 0$ and solve for t. □

The proposition's hypothesis is the interesting part, because on a looped district it is false, and the example is arranged so that the reader can see by how much.

The cascade. At $t = 0$ the transfer main is lost. Nothing is starved: the two reservoirs take over, and the first thing to ask is how they share the load. That is not a matter of their size but of the district's hydraulics, and the script solves for it. With both full, reservoir 5 supplies 9.79 kg/s and reservoir 6 supplies 11.21, which sum to the district's whole 21. Reservoir 5 is holding 420 tonnes and delivering 47 percent of the district; reservoir 6 is holding two and a half times as much and delivering 53. The small reservoir is being asked to do almost half the work, and by Proposition 16.4 at that draw it would reach its minimum in 11.9 hours.

It does not. As reservoir 5's level falls below reservoir 6's, the loop between junctions 5 and 6 begins to carry water the other way, and reservoir 6 takes over part of reservoir 5's share. The draw on reservoir 5 drifts from 9.79 kg/s to 7.21 over the interval, and the level curve bends. Integrating (16.9) on both levels by implicit Euler, with a steady solve of the district inside every step and the event located by bisection on the interpolated state, puts the first event at 14.4 hours rather than 11.9. The closed form was wrong by twenty-one percent, and it was wrong in the safe direction only by accident: had reservoir 5 been the larger of the two the coupling would have run the other way. The accumulation term made the steady solve a function of the

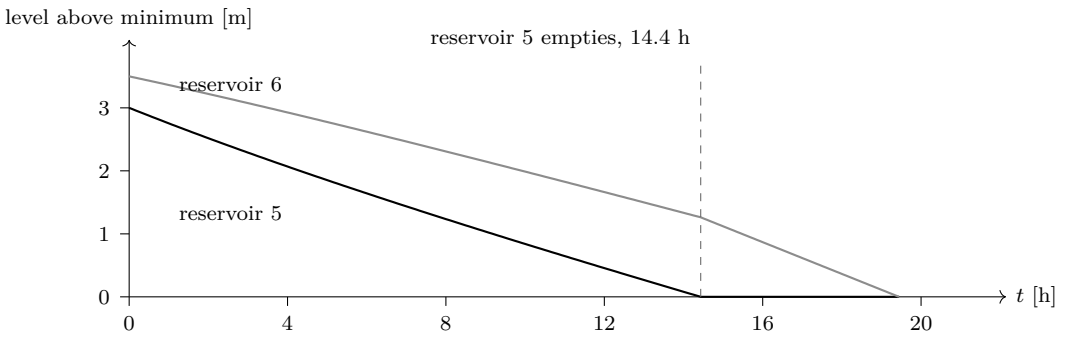


Figure 16.4: Levels of the two service reservoirs after the loss of the transfer main at $t = 0$: reservoir 5 (black) and reservoir 6 (grey). Both curves bend as the falling levels shift the share between them. Reservoir 5 reaches its minimum at 14.4 hours and shuts; reservoir 6's fall then steepens as it takes the district's whole demand, and the district is dry at 19.4.

state, which is the whole content of this chapter, and the price of forgetting it is two and a half hours of an operator's planning.

At 14.4 hours reservoir 5 reaches its minimum operating level and its outlet valve shuts. Reservoir 6, which by then is at 1.26 m of its 3.5 having given up 671 cubic metres, is left carrying the district alone: the whole 21 kg/s, nearly twice what it had been supplying. District B is dry at 19.4 hours, five hours after the first event. Had reservoir 6 gone on carrying only its own 11.2 kg/s it would have lasted 26.0 hours; the cascade cost six and a half.

Figure 16.4 draws the two levels. The operator's window is the 14.4 hours to the first event, and the deadline is the 19.4 to the second. A steady-state analysis says district B is fed from storage; the dynamic one says for how long, and says that the deadline is not the sum of two independent drainings but shorter than either reservoir would suggest alone.

How long, with error bars. The district's demand is a forecast, not a measurement, and on a distribution network it is not known to better than thirty percent. Draw it from a log-normal with that spread and the window has a distribution: a median of 15.1 hours, a fifth percentile of 9.4, a ninety-fifth of 24.6, and a 7.8 percent chance that the window is shorter than ten hours. That is the honest form of "how long do I have": a time with a tail, and the tail is what the operator plans against. Note that the uncertainty is one-sided in its consequences and two-sided in its arithmetic — a window that turns out to be twenty-four hours costs nothing, one that turns out to be nine costs a district.

> **Principle 16.4.** *A cascade is a sequence of events located on a trajectory, each an intervention caused by the last. The steady state says what is drawing on what; the dynamics say how long until it runs out, and the uncertainty in the dynamics says how much of that time can be counted on.*

The window as a decision. The window is worth something only if an action inside it changes the outcome, and the graph says which action does. Curtailing demand at junction 2 does nothing whatever: with the transfer main gone, district A and district B are two disconnected networks, and no demand in A passes through B. This is the sharper form of the lesson a meshed

Table 16.4: Curtailing district B's demand pro rata at the moment the transfer main is lost: the window to reservoir 5's minimum level, and the time at which the district runs out of water altogether. Curtailing in district A, which is now disconnected, changes neither.

curtailment [kg/s]	window [h]	district dry at [h]
0	14.44	19.44
2	16.64	21.49
4	19.42	24.02
6	22.99	27.22
8	27.63	31.41

network only hints at — the action must be taken inside the component that is draining, and after a bridge fails the component is smaller than the operator's habit assumes.

Curtail district B's own demand, pro rata across its three junctions, and Table 16.4 gives the result. Two kilograms a second, a tenth of the district, buys two and a quarter hours; four buys five; eight buys thirteen. The return is better than proportional, because the reduction acts on the denominator of (16.10), and every hour bought is an hour in which a tanker, a repair or a temporary connection can arrive.

16.11 In practice

Integrate implicitly, step for the slow dynamics. The systems of this book couple light components to heavy ones and are stiff by construction. An implicit method with a step chosen for the dynamics of interest is the default; an explicit method is right only when the fastest time constant is the one of interest, which is rare. Let the step control shrink the step at events and transients and grow it when the state settles.

Carry the storage states; recover the rest. The filter's state is the set of storage states and the random-walk inputs, and nothing else, by Proposition 16.2. Heads at zero-storage nodes and every flow are algebraic functions of the state and are recovered by a solve when asked for. Carrying them in the state is a common mistake and it makes the covariance larger and singular.

Set the process noise from the physics, then check it. The width of a storage row is the model's error over one step; the width of a random walk is how far the input may move in one step. Both are physical statements and should be set as such, then checked by the innovations: a filter whose standardized innovations drift, as in §16.6 during the ramp, has too little process noise on the input that is moving; one whose innovations are too small has too much and is tracking noise.

Smooth when you can, filter when you must. Operation in real time filters; every analysis after the fact should smooth, since the smoother has the same cost and less error. A common mistake is to report filtered estimates of a past state when the later readings were available.

Locate events; do not step over them. An integrator that steps past a limit crossing and applies the change at the end of the step has mis-timed the event by up to a step, and in a

cascade the timing is the answer. Locate the crossing by bisection, apply the change there, and restart the step.

Report a window, not a time. The time to a trip depends on quantities, wind and room among them, that are not known precisely. Report the window as a distribution, and plan against its short tail.

16.12 What is missing

Every model in this chapter was one system. The unrolled graph is a chain of copies of it, and the chapter organized computation along the chain. Chapter 17 organizes it across the other direction, the regions of Chapter 8 arranged in a tree of subsystems, and asks what it means for a region to be summarized to its parent when both have dynamics; and Chapter 18 asks how to cut a large system into such regions, with the measurements that Chapter 5 and Chapter 9 promised.

Notes and further reading

Differential-algebraic systems, their index, and the backward differentiation formulas for stiff problems are treated by Brenan, Campbell and Petzold [12] and by Hairer and Wanner [32], who also give the stability analysis behind Table 16.1; the step-change controller mentioned in §16.2 follows the PI design of Gustafsson, Lundh and Söderlind [29]. That a BDF step is the steady system with its storage rows rewritten, so that simulation is a loop over one solver, is the book's framing of a standard fact.

The Kalman filter is Kalman's [38] and the fixed-interval smoother is Rauch, Tung and Striebel's [75]; Särkkä and Svensson [82] present both as Gaussian inference on a chain, which is the view of §16.5, and treat process noise, random-walk models and innovation diagnostics. The derivation of the predict step as a Schur complement and of the update as the rank-one conditioning of Chapter 11 is the book's, and it is the reason the filter needed no new machinery.

The reservoir model of §16.10 is the standard one: a tank of constant plan area whose level is a state variable and whose balance is its net inflow, as in the EPANET extended-period simulation that most water utilities run [79]. Extended-period simulation is exactly the construction of §16.2 — a sequence of steady hydraulic solves with the tank levels carried between them — usually with explicit Euler and a fixed hydraulic time step rather than the implicit step used here. What the fixed step costs is precisely the two and a half hours the closed form got wrong: an explicit step evaluates the draw at the level it starts from, and the error accumulates in the same direction throughout. The practice of running a district from storage after losing its supply is *intermittent supply* planning, and the sharing of a demand between two floating reservoirs is the classical multiple-source problem that the gradient method of Todini and Pilati [93] solves inside each step. The event-driven view of §16.9, in which each valve closure is an intervention on the model that the next integration starts from, is how the reference implementation of Appendix C handles it.

Summary

- With storage, the model is an index-one differential-algebraic system; the graph gains one variable and one square per storing node.
- A BDF step is the steady system with its storage rows rewritten: same variables, same sparsity, same solver. Implicit methods are stable at any step; the explicit method fails above $2/|\lambda_{\max}|$, 1.41 seconds on the chain, and is accurate to 10^{76} metres at two.
- The unrolled graph is a chain; the storage states separate past from future; the precision is block-tridiagonal and factors in linear time.
- Process noise is the storage row's width; a random walk on an input tracks its drift.
- Filtering, smoothing and prediction are elimination orders on the chain. The Kalman predict step is a Schur complement and the update is a rank-one conditioning. On the chain the smoother halves the filter's errors and the batch solve agrees with it to 10^{-9}.
- The innovations in time are the model's diagnostic: white when it is adequate, biased when it lags, spiked when the system changed. A source drop of a quarter is six spreads on the first reading after it.
- A source drop and a sensor bias with the same steady effect are one hypothesis in steady state and two in time; the evidence, one innovation density per reading, separates them by 128 nats within the time constant and not at all after it.
- A cascade is events on a trajectory. After the loss of the transfer main, the smaller service reservoir reaches its minimum level in 14.4 hours and its closure leaves the larger one carrying the district alone; with the demand uncertain by thirty percent the window has a median of 15.1 hours and a fifth percentile of 9.4.
- The window is a decision when an action inside it changes the outcome. Curtailing in the district the bridge used to feed from does nothing, because the two districts are now disconnected; curtailing 4 kg/s of district B's own 21 moves the deadline from 19.4 hours to 24.0.

Exercises

Exercise 16.1. Derive the BDF2 formula (16.5) by fitting a quadratic through X^{n-1}, X^n and X^{n+1} at equal spacing and requiring its derivative at t_{n+1} to equal $\mathcal{B}(\mathbf{z}^{n+1})$. Show that on $\dot{X} = \lambda X$ the method's amplification factor has magnitude below one for every $\lambda < 0$ and every Δt.

Exercise 16.2. Complete the proof of Proposition 16.3: show that the Schur complement of the joint precision onto $\mathbf{x}^{n+1}$ equals $(\mathbf{Q} + \mathbf{F}\boldsymbol{\Sigma}_n\mathbf{F}^{\mathsf{T}})^{-1}$ by the Woodbury identity, and derive the predicted mean $\mathbf{F}\boldsymbol{\mu}_n + \mathbf{c}$ from the information vector.

Exercise 16.3. Rerun the tracking example with the random-walk width raised to 3 kg/s per step and lowered to 0.3. Report the filter's error and the mean standardized innovation during the ramp for each, and explain the trade-off between lag and noise that the width controls.

Exercise 16.4. In the cascade, suppose the operator, instead of curtailing across the whole district, closes the zone valve to junction 6 at $t = 6$ hours, isolating its 6 kg/s of demand on purpose. Recompute the draws on the two reservoirs from their levels at that moment, the time at which reservoir 5 reaches its minimum, and whether the cascade still happens. Compare the

demand lost by this action with the 4 kg/s of Table 16.4 and with the 21 the district loses when it runs dry, and say which the operator should prefer.

Exercise 16.5. Give reservoir 5 a level gauge reading every ten minutes with accuracy 2 cm, and build the filter for its level with the district's demand as a random-walk parameter. Show that after three readings the filter's predicted time to the minimum level has a spread narrower than the 9.4-to-24.6-hour window of §16.10, and say why a level gauge is worth more here than a flow meter on the transfer main.

Solutions to selected exercises

Exercise 16.1. Put $t_{n+1} = 0$ and let the quadratic through $(-2\Delta t, X^{n-1})$, $(-\Delta t, X^{n})$, $(0, X^{n+1})$ be $p(s) = X^{n+1} + as + bs^2$. Then $X^n = X^{n+1} - a\Delta t + b\Delta t^2$ and $X^{n-1} = X^{n+1} - 2a\Delta t + 4b\Delta t^2$. Subtracting twice the first from the second gives $X^{n-1} - 2X^n = -X^{n+1} + 2b\Delta t^2$, so $b = (X^{n+1} - 2X^n + X^{n-1})/(2\Delta t^2)$, and then $a = (X^{n+1} - X^n)/\Delta t + b\Delta t = (3X^{n+1} - 4X^n + X^{n-1})/(2\Delta t)$. The derivative at $s = 0$ is a, and setting $a = \mathcal{B}(\mathbf{z}^{n+1})$ gives $3X^{n+1} - 4X^n + X^{n-1} = 2\Delta t\,\mathcal{B}(\mathbf{z}^{n+1})$, which is equation (16.5). On $\dot X = \lambda X$ with $z = \lambda\Delta t$ the recurrence is $(3-2z)X^{n+1} = 4X^n - X^{n-1}$, whose characteristic roots satisfy $(3-2z)r^2 - 4r + 1 = 0$; for $z < 0$ the product of the roots is $1/(3-2z) < 1/3$ and their sum is $4/(3-2z) < 4/3$, and one checks that both roots lie inside the unit disk for every $z < 0$, with the larger root $\to 1$ as $z \to 0^-$ and both roots $\to 0$ as $z \to -\infty$: the method damps every stable mode at every step, and damps the stiffest modes most.

Exercise 16.2. Write $\mathbf{S} = \boldsymbol{\Sigma}_n^{-1} + \mathbf{F}^{\mathsf T}\mathbf{Q}^{-1}\mathbf{F}$. The Schur complement is $\mathbf{Q}^{-1} - \mathbf{Q}^{-1}\mathbf{F}\mathbf{S}^{-1}\mathbf{F}^{\mathsf T}\mathbf{Q}^{-1}$. The Woodbury identity states $(\mathbf{Q} + \mathbf{F}\boldsymbol{\Sigma}_n\mathbf{F}^{\mathsf T})^{-1} = \mathbf{Q}^{-1} - \mathbf{Q}^{-1}\mathbf{F}(\boldsymbol{\Sigma}_n^{-1} + \mathbf{F}^{\mathsf T}\mathbf{Q}^{-1}\mathbf{F})^{-1}\mathbf{F}^{\mathsf T}\mathbf{Q}^{-1}$, which is the same expression, so the predicted precision is $(\mathbf{Q} + \mathbf{F}\boldsymbol{\Sigma}_n\mathbf{F}^{\mathsf T})^{-1}$ and the predicted covariance is $\mathbf{F}\boldsymbol{\Sigma}_n\mathbf{F}^{\mathsf T} + \mathbf{Q}$. For the mean, the joint information vector has blocks $\boldsymbol{\Sigma}_n^{-1}\boldsymbol{\mu}_n - \mathbf{F}^{\mathsf T}\mathbf{Q}^{-1}\mathbf{c}$ on $\mathbf{x}^n$ and $\mathbf{Q}^{-1}\mathbf{c}$ on $\mathbf{x}^{n+1}$; the Schur-complement information vector is $\mathbf{Q}^{-1}\mathbf{c} + \mathbf{Q}^{-1}\mathbf{F}\mathbf{S}^{-1}(\boldsymbol{\Sigma}_n^{-1}\boldsymbol{\mu}_n - \mathbf{F}^{\mathsf T}\mathbf{Q}^{-1}\mathbf{c})$. Multiplying by the predicted covariance and simplifying with $\mathbf{S}^{-1}\boldsymbol{\Sigma}_n^{-1} = \mathbf{I} - \mathbf{S}^{-1}\mathbf{F}^{\mathsf T}\mathbf{Q}^{-1}\mathbf{F}$ gives $\mathbf{F}\boldsymbol{\mu}_n + \mathbf{c}$; the reader who does not want the algebra can verify it on the scalar case, where every matrix is a number and the identity is one line.

Chapter 17

Hierarchy as physical decomposition

Every engineered system of any size is described at several levels at once. A supply zone is one offtake to the trunk network, a set of district metered areas to the zone's own telemetry, a set of sub-zones to each district, and a set of service connections to each sub-zone; a metered district is one demand to the network above it and a tree of draws to itself; a chemical plant is units to the site and vessels to each unit. The levels are not a convenience of drawing. Each is a boundary-exchanging system in the sense of Chapter 1, with its own balance rows and closures and its own ports, and the levels nest because the cuts nest: the boundary of a sub-zone lies inside the boundary of its district, which lies inside the boundary of the zone.

Chapter 8 showed what one cut buys. A region's factors, eliminated onto its ports, are the region as its neighbors see it, exactly. This chapter applies that statement to nested cuts, and finds that it needs nothing new. The same conservation law that makes the flows across a cut exact makes a region's balance the sum of its parts' balances, so that a parent's model of a child is the child's message, and a grandparent's model of the child is the child's message passed through the parent's. Schur complements compose, so the hierarchy of regions is a junction tree whose messages are computed level by level and reused across levels. And the cut that organizes the model organizes control and scale as well: a setpoint is imposed at a port, a child meets it inside, and the cost of a local change is the cost of one path from leaf to root.

17.1 One conservation law used twice

Take a region S of a physical graph: a set of nodes, the edges among them, and the sources at them. Chapter 1's Proposition 1.1 said that in steady state the net flow across the boundary of S equals the net source inside S. Write it as an equation among the residual rows. The balance at node n is

$$r_n \;=\; \sum_{e \ni n} \pm F_e \;-\; q_n \;=\; 0, \tag{17.1}$$

with the sign by the edge's orientation at n. Sum the rows over $n \in S$. Every edge with both ends in S appears twice with opposite signs and cancels; every edge with one end in S appears once. Hence

$$\sum_{n \in S} r_n \;=\; \sum_{e \in \partial S} \pm F_e \;-\; \sum_{n \in S} q_n, \tag{17.2}$$

which is the balance row of S regarded as a single node with the cut edges as its edges and the total source as its source. This is the whole content of the proposition, and it has a consequence

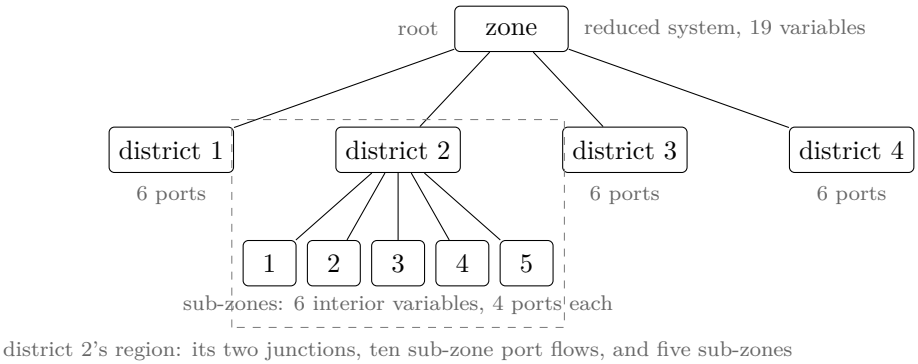


Figure 17.1: The supply zone's hierarchy of regions. Each sub-zone eliminates six interior variables onto four ports; each district eliminates its own twelve (its two junction heads and the ten sub-zone port flows) onto six; the zone solves the reduced system of nineteen. Every edge of the tree is a cut of the physical graph, and the ports of a cut are the message that crosses it.

for modeling that is easy to pass over.

Proposition 17.1 (Aggregation)**.** *Let a region S be replaced, in the parent's model, by one node whose edges are the cut edges of S and whose source is $\sum_{n\in S} q_n$. The parent's balance row at that node is the sum of the region's balance rows. It is exact, and it reads only the cut flows and the total source.*

The proposition is the license for every meter that reads a total. A district meter reads the district's total demand; the network operator's meter at a zone inlet reads the zone's total draw; a building's utility meter reads the sum of everything inside. None of these instruments knows the inside, and none needs to: equation (17.2) says that the total draw is a function of the cut flows alone, so a factor on the total is a factor on the ports. A meter on a district's demand is, to the zone, a factor on the district's port flows; the zone can use it without a model of the district. Where the meter is placed in the hierarchy is therefore a choice, and Section 17.4 shows the two placements agree to the physics width.

What the proposition does not license is as important, and Chapter 1 said it: the intensive states do not aggregate. The region-as-node has one balance row and no head. If the parent's model gives it a head, that head is a modeling assumption, not a consequence of conservation, and Section 17.3 shows what the assumption costs. The exact object that replaces the region is not a node with a potential but the region's message, which carries as many potentials as the region has ports.

17.2 Nested messages

Proposition 8.2 eliminated one region's interior onto its ports. In a hierarchy the interior of a district contains the interiors of its sub-zones, and there are two ways to eliminate it: all at once, or sub-zone by sub-zone and then the rest. The two agree, because the Schur complement has a composition property that is the algebraic form of nested cuts.

Proposition 17.2 (Schur complements compose)**.** *Let $\mathbf{\Lambda}$ be a precision on variables partitioned as $I_1 \cup I_2 \cup S$, with I_1 and I_2 disjoint. Eliminating I_1 first and then I_2 from the result gives the*

same precision on S as eliminating $I_1 \cup I_2$ at once:

$$(\mathbf{\Lambda}/\mathbf{\Lambda}_{I_1 I_1})/(\mathbf{\Lambda}/\mathbf{\Lambda}_{I_1 I_1})_{I_2 I_2} \;=\; \mathbf{\Lambda}/\mathbf{\Lambda}_{II}, \qquad I = I_1 \cup I_2, \tag{17.3}$$

where $\mathbf{\Lambda}/\mathbf{\Lambda}_{AA}$ denotes the Schur complement of the block A. The same holds for the information vector.

Proof. Both sides are the precision of the marginal on S of the Gaussian with precision $\mathbf{\Lambda}$. The left integrates out I_1 and then I_2; the right integrates out both together. Marginalization is one integral whatever the order in which its variables are grouped, so the two marginals are the same Gaussian and have the same precision and information. In matrix terms the identity is the quotient formula for Schur complements, and it can be verified by block elimination: eliminating I_1 leaves a bordered system on $I_2 \cup S$ whose I_2 block is $\mathbf{\Lambda}_{I_2 I_2} - \mathbf{\Lambda}_{I_2 I_1}\mathbf{\Lambda}_{I_1 I_1}^{-1}\mathbf{\Lambda}_{I_1 I_2}$, and eliminating that block reproduces the 2×2 block inverse of $\mathbf{\Lambda}_{II}$ term by term. □

The proposition is why a hierarchy needs no new machinery. Take the zone. Each sub-zone's factors read its interior and its four ports; its message is a 4×4 precision and a 4-vector on those ports, computed by equation (8.5). The district's factors read the district's two junctions, the sub-zone port flows, and the district's own ports. Add the five sub-zone messages to the district's precision, and the district's interior at this level is twelve variables, its junctions and the ten sub-zone port flows, with the sub-zone interiors already gone. Eliminate the twelve onto the district's six ports. By Proposition 17.2 the result is the message the district would have sent had its forty-two variables been eliminated at once, and it was computed with five 6×6 solves and one 12×12 solve instead of one 42×42. Add the four district messages to the zone's factors and solve the reduced system on the nineteen variables the zone reads. Then back-substitute downward: Proposition 8.3 gives each district's interior from its ports, and, applied again with the district's interior as the sub-zone's ports, each sub-zone's interior from the district's. Figure 17.1 draws the tree.

The pattern generalizes. A hierarchy of regions is a junction tree in which each region's cluster is its own interior together with its ports, and the separators are the ports. Messages go up the tree by Schur complement and down by back-substitution, and the running intersection property of Chapter 8 holds because a variable shared by two regions is a port of the lower one and lies on the path between them. The cost model of Section 8.5 applies level by level: a region with n_r variables eliminated at its level and k_r ports costs one factorization of an $n_r \times n_r$ block and k_r solves against it, and the root costs a solve of the size of what it reads. What the hierarchy adds to the flat partition of Chapter 8 is that n_r at each level is the region's own variables, not the whole subtree's: the sub-zone interiors are eliminated once, at the sub-zone level, and every level above sees them only through four numbers per sub-zone.

Principle 17.1. *Nested cuts give nested messages. A region's message to its parent is computed from its children's messages and its own factors, and it is the same message the region would send if it were eliminated whole. Each level pays for its own variables only.*

17.3 What the parent needs to know

A level of description is a decision about what to carry. The hierarchy makes the decision precise: the parent needs, of each child, exactly the child's message, a precision and an information vector on the child's ports. Nothing less is exact and nothing more is needed. Two readings of that message are useful.

The message is an equivalent network. On the district's six ports the message is a 6×6 precision $\mathbf{S}$ and a 6-vector $\mathbf{h}$. Order the ports as the four flows $\mathbf{f}$ and the two potentials $\boldsymbol{\phi}$, and partition. The Gaussian with precision $\mathbf{S}$ has, conditional on the potentials, a mean and covariance for the flows of

$$\mathbb{E}[\mathbf{f} \mid \boldsymbol{\phi}] \;=\; \mathbf{S}_{ff}^{-1}\mathbf{h}_f \;-\; \mathbf{S}_{ff}^{-1}\mathbf{S}_{f\phi}\,\boldsymbol{\phi}, \qquad \mathrm{Cov}(\mathbf{f} \mid \boldsymbol{\phi}) \;=\; \mathbf{S}_{ff}^{-1}, \tag{17.4}$$

which is a network: a matrix of equivalent conductances $-\mathbf{S}_{ff}^{-1}\mathbf{S}_{f\phi}$ from the port potentials to the port flows, an equivalent source $\mathbf{S}_{ff}^{-1}\mathbf{h}_f$, and an error bar $\mathbf{S}_{ff}^{-1}$ on the flows. This is the Kron reduction of Chapter 8 with uncertainty, and it is what a modeler who knows only the ports would have to build by hand. For district 0 of the supply zone the equivalent conductance from the trunk junction's head to the district's flow to the trunk junction is -0.58 kilograms a second per metre, where the physical pipe alone has 5: the message knows that the district's outlet junction has a logger, so that raising the trunk junction's head mostly raises the outlet junction with it rather than reversing the flow. The flows' spreads given the heads are 0.04, 0.01, 0.33 and 0.03 kilograms a second. A network model that treats a supply zone as "a demand of 129 kilograms a second at 66 metres" has built a one-node equivalent; the message is the many-port equivalent, with the region's readings folded in, and Section 17.4 measures the difference.

The message is the parent's whole knowledge of the child. Whatever the parent asks about the child's interior, it asks by back-substitution, and back-substitution needs the child's factorization, which the child keeps. The parent does not hold the child's variables. This is the level-of-description discipline enforced by the algebra: the zone's reduced system has nineteen variables, not one hundred eighty-seven, and the zone answers questions about sub-zones by passing them down.

Lumping is the message with an assumption added. The traditional alternative to the message is to *lump*: replace the child by a node with one potential and the total source, as Proposition 17.1 allows for the balance and forbids for the potential. Lumping is exact for the flow through the cut and wrong for everything that depends on which port the flow used. In the district, water enters at the inlet junction near seventy metres and leaves at the outlet junction near sixty; a node at one head cannot do both, and the parent's estimate of which way the district's flow went is wrong by the difference. The structural literature knows this as the difference between Guyan reduction, which is the Schur complement onto the retained degrees of freedom and is exact for the static response [30], and a lumped mass, which is not. The message needs no judgment about which potential to keep; it keeps them all, and there are only as many as the cut has edges.

> **Principle 17.2.** *The parent's model of a child is the child's message: exact on the ports, silent on the interior, and answered for by back-substitution when the interior is asked about. A lumped node is the message with one potential where the cut has several, and it is exact for the total flow and wrong for its distribution.*

17.4 Worked example: a supply zone in three levels

Figure 17.2 draws the zone. Water is supplied from a service reservoir whose top water level, 72 metres, the model treats as the reference. Each of four district metered areas has an inlet junction

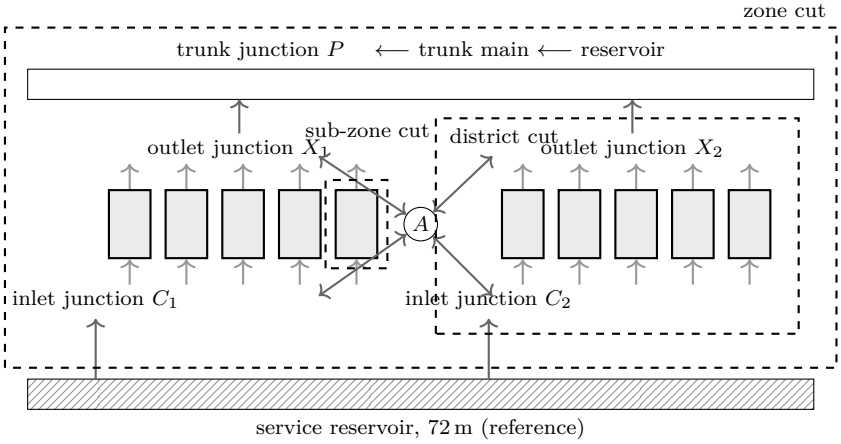


Figure 17.2: The supply zone, two of its four districts drawn. Water enters at the inlet junctions, passes through the sub-zones (inlet node I, demand node D where the draw is taken, outlet node O), collects at the outlet junctions, which the trunk junction P feeds from the reservoir; the ring junction A exchanges with every district junction. The dashed rectangles are three nested cuts. A sub-zone's ports are its inlet and outlet flows with the district's heads on the far side; a district's are its four exchange flows with P and A; the zone's is the exchange with the service reservoir.

C, from which five sub-zones are fed, and an outlet junction X, into which they discharge; the outlet junction is fed again from a trunk junction P that the reservoir supplies down the trunk main, and both junctions of every district exchange with a ring junction A that stands for the zone's cross-connections. A sub-zone has an inlet node, a demand node where the draw is taken, and an outlet node; most of what enters is drawn off, and the rest passes on. Every pipe is a conductance, and the model is the linear-Gaussian one of Chapter 6: one balance row per junction, one closure per pipe, both with a physics width of ten grammes a second; a prior of 8 ± 1.5 kilograms a second on each sub-zone's demand; pressure loggers on every district junction, on the trunk junction and on the ring junction with accuracy half a metre; outlet loggers on two sub-zones per district with accuracy one metre; and a flow meter on each district with accuracy 0.4 kilograms a second. The model has 187 variables, seventy heads, ninety-seven flows and twenty demands, and 209 factors. The truth draws the twenty demands from their prior, and the readings are the truth's values plus noise. Every number is from `ch17_hierarchy.py`.

Head loss is a conductance times a head difference here, as in Chapter 8: the linearization about an operating point, which is what lets the whole chapter's algebra be exact.

Three solves agree. The monolithic solve inverts the 187×187 precision, whose condition number is 1.4×10^8 in these units. The hierarchical solve computes twenty sub-zone messages, four district messages, one 19-variable zone solve, and back-substitutes; Table 17.1 gives the sizes. The two agree to 2×10^{-5} in every mean and 6×10^{-11} in every standard deviation, which is Proposition 17.2 in floating point. A third solve writes each district's meter on the district's five demands instead of on its four port flows, the two placements Section 17.1 said are equivalent, and agrees with the first to 0.9 grammes a second, the physics width.

Table 17.1: The levels of the supply zone: how many regions at each, how many variables each eliminates at its own level, and how many ports it keeps.

level	regions	eliminated per region	ports	what the level reads
sub-zone	20	6	4	inlet, demand node, outlet, demand, two flows
district	4	12	6	two junctions, ten sub-zone port flows
zone	1	reduced system of 19	–	P, A, trunk flow, sixteen port flows

Table 17.2: Posterior spreads in the supply zone as the readings of each level are admitted, from the root down. Demands and flows in kilograms a second.

readings admitted	demand, with outlet logger	demand, without	district 1 total	trunk flow
none (priors)	1.50	1.50	3.35	5.58
zone loggers (P, A)	1.48	1.48	3.17	4.20
+ district loggers and meters	1.34	1.34	0.39	0.64
+ sub-zone outlet loggers	0.58	1.26	0.39	0.63

What the posteriors say. The truth has district inlets at 69.3 to 69.5 metres, district outlets at 58.5 to 59.5, the trunk junction at 65.6, and demand-node heads from 31 to 48. The posterior of the trunk junction is 65.57 ± 0.03 and of the flow down the trunk main 128.6 ± 0.6 kilograms a second against a truth of 128.8. District 0's flow through its cut, the combination $f_s + f_{ai} - f_p - f_{ao}$ of its four port flows, is 36.93 ± 0.39 kilograms a second, and the sum of its five demand posteriors is 36.93, the same number, as equation (17.2) requires. Inside the sub-zones the picture is what Chapter 12 would predict: a sub-zone with an outlet logger has its demand node at 38.8 ± 1.6 metres and its demand at 8.01 ± 0.58 kilograms a second (truth 38.6 and 8.03); a sub-zone without one has 41.9 ± 3.4 and 6.84 ± 1.26 (truth 39.6 and 7.67), the district meter and the junction loggers having narrowed its demand from the prior's 1.5 but the head remaining a virtual sensor two levels below any reading.

Information by level. Table 17.2 asks which readings narrow what, by solving the zone with the readings of each level admitted in turn. The zone's two loggers, on the trunk junction and the ring junction, narrow the trunk flow from 5.6 to 4.2 kilograms a second and the sub-zone demands hardly at all: two readings at the root cannot say much about twenty demands three levels down. The district's loggers and meters narrow the district total from 3.2 to 0.39 kilograms a second, which is the meter's accuracy, and the trunk flow to 0.64, but leave each demand at 1.34: the meter pins the sum of five demands and says nothing about their split, so the five posteriors are negatively correlated and each stays wide. Only the sub-zone's own outlet logger narrows its demand, to 0.58, and it does nothing for the sub-zone next to it. Information enters at a level and flows through the ports; what a port carries is what a level can learn from above, and a port that carries a total carries a total.

Uncertainty through two levels. Equation (8.6) splits a region's interior covariance into two terms, the region's own conditional uncertainty given its ports and the port uncertainty carried in by the sensitivity $\mathbf{X}$; in a hierarchy the second term is itself the sum of the district's own conditional term and what the district's ports carried down from the zone, by the same equation applied one level up. For the unlogged sub-zone the split is stark. Given its four ports, the inlet and outlet flows and the district's two heads, its demand node is known to 0.02 metres

Table 17.3: The exact hierarchical posterior against the lumped zone, in which each district is one node. Flows in kilograms a second, heads in metres.

quantity	exact	lumped	true
trunk junction head P	65.57 ± 0.03	68.13 ± 0.02	65.56
flow down the trunk main	128.6 ± 0.6	77.3 ± 0.4	128.8
district 0 flow to P, f_p	-30.7 ± 0.3	-18.3 ± 0.2	-30.4
district 0 total demand	36.9 ± 0.4	36.9 ± 0.4	36.6
district 0 head(s)	69.5 / 59.5	64.5	69.5 / 59.5
misfit on the readings	24.6 / 22	1 034 / 14	–

and its demand to 0.02 kilograms a second: the sub-zone's own factors are physics rows of width ten grammes a second, and once what crosses its boundary is fixed, conservation fixes the inside. All of its 3.4 metres and 1.26 kilograms a second are carried in through the ports, almost entirely through the outlet flow, whose posterior spread is 1.16 kilograms a second. Principle 1.3 said the boundary is where a model's ignorance concentrates; the hierarchy makes the statement quantitative at every level, because each level's uncertainty about a child is a term in the child's covariance, and the terms can be read off separately.

The lumped zone. Now replace each district by one node L_i with the district's total demand, the district's four exchange edges, and the district's two loggers both reading L_i. Table 17.3 compares. The lumped model gets the district totals right, to the meter's accuracy, because Proposition 17.1 guarantees it. It gets the trunk junction wrong by 2.6 metres and the trunk flow wrong by 51 kilograms a second, because its one node at 64.5 metres draws 19 kilograms a second per district straight from the reservoir through the inlet main, where the true inlet junction at 69.5 draws six. The reported spreads are 0.02 metres and 0.4 kilograms a second: the lumped model is confidently wrong, in the sense of Chapter 4, and its misfit says so, 1 034 on fourteen readings against the exact model's 24.6 on twenty-two. Two loggers ten metres apart on one node cannot both be fitted. A modeler could lump more carefully, with two nodes per district, and would do better; the message does not require the judgment, because it keeps every port.

A local change. Add an outlet logger to sub-zone $(1, 3)$, which had none. Its demand-node posterior moves from 41.9 ± 3.4 to 39.9 ± 1.6 and its demand from 6.84 ± 1.26 to 7.60 ± 0.58, on a truth of 7.67. The largest change of any posterior mean is 2.04 inside that sub-zone, 0.82 in the other sub-zones of its district, whose demands share the district meter with it and are explained away, 0.008 in its district's junctions, 0.005 in the sub-zones of the other districts, and 0.002 in the zone's junctions. The hierarchy recomputes exactly what moved: the sub-zone's message, a 6×6 elimination; its district's message, a 12×12; and the zone solve of nineteen. Nineteen sub-zone messages and three district messages are reused. The monolithic route refactorizes the 187×187 precision.

Figure 17.3 prices the two routes as the zone grows, and the price is worth reading carefully, because the clock and the arithmetic disagree.

On the clock the hierarchical update wins, and by a widening margin. At 187 variables the monolithic sparse solver is faster, a third of a millisecond against about half of one. At 2 659 variables the hierarchical route is ahead by a factor of six, and at 25 987 it takes about three

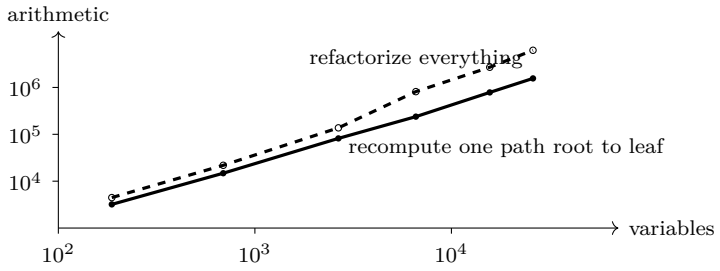


Figure 17.3: Arithmetic cost of incorporating one new logger in one sub-zone, as the zone grows from four districts of five sub-zones to sixty-four of fifty. Filled: nonzeros in the sparse factor of the whole precision. Open: the flops of the recomputed eliminations, the changed sub-zone's message, its district's message and the zone solve, with every other message held. Both axes logarithmic. The counts are plotted rather than timings because a committed figure has to be identical on re-render, and because they tell a different story from the clock: see the text.

milliseconds against a hundred and sixty: a factor of fifty. Those are wall-clock figures from one machine, best of three runs, and they are not part of what this book commits: re-run them and they move.

The arithmetic says something else. The sparse factor of the whole precision grows from 3 199 nonzeros to 1.6 million, a factor of 490; the flops of the hierarchical update grow from 4 470 to 6.2 million, a factor of 1 390. *The update does more arithmetic than the refactorization at every size past the smallest, and its advantage in arithmetic is shrinking, not growing.* The reason is in the third term of the update. The sub-zone's message and the district's message are fixed in size, 6×6 and 12×12, whatever the zone does; but the zone solve is a dense system of $3 + 4R$ variables, 19 at four districts and 259 at sixty-four, and its cost grows as R^3. The update's cost is therefore *not* fixed by the path from the changed sub-zone to the root, as it is tempting to say. Only the path's interior levels are fixed. The root grows.

What the clock is measuring is a constant, not an exponent: a dense 259×259 solve runs at a large fraction of the machine's peak, while a sparse factorization of 26 000 variables spends most of its time on indices. The measured factor of fifty is real and it is worth having, but it is a statement about two implementations, and it would turn against the hierarchy at a zone large enough for the root solve to dominate. The remedy is not a faster solver: it is to cut the root as well, which is what Chapter 18 does. Both routes give the same answer, to 7×10^{-9} at the size where the check was run. The sparse solver's factorization is itself a nested elimination, chosen by its ordering heuristic, so the hierarchy does not beat it at solving; what the hierarchy adds is the reuse, which a solver that does not know the physics cannot offer.

17.5 The cut, three times

The hierarchy of cuts has served so far as a way to organize inference. It is the same hierarchy that organizes two other things engineers do with large systems, and the coincidence is not one.

Modeling. A level of description is a choice of cut. The zone's operator models districts, the district's designer models sub-zones, the sub-zone's designer models service connections; each draws the boundary that the questions of that level need and treats what is inside as a message

and what is outside as a reservoir or a port. Principle 1.3 said a system is defined by what crosses its boundary; the hierarchy says a system is defined at every level by that level's boundary, and the levels agree because the messages compose.

Control. A controller acts at a port. A pumping station sets the reservoir's outlet head, which is the zone's boundary condition; a distribution operator asks a district to hold its inlet flow; a regulator gives a supply zone an abstraction limit. Each is an intervention in Chapter 13's sense, a named quantity imposed at a cut, and each is met by the child, which back-substitutes the imposed port value into its interior and finds what its own inputs must do. The parent needs the child's message to know how the port responds to what it imposes, and nothing else. This is the structure of hierarchical control as Mesarović, Macko and Takahara set it out [60]: coordination at the interfaces, local decisions inside, the interfaces being exactly the ports. Section 17.6 works one: a district metered area under a distribution junction.

Scale. A large system is solved as a tree of regions, and the cost is set by the ports, not by the interiors. The cuts that give a good hierarchy for inference are those with few ports relative to their interiors, and the physical hierarchy, which was drawn for modeling and control, tends to have that property, because engineers build subsystems with narrow interfaces on purpose. Chapter 18 takes this up and measures it.

> **Principle 17.3.** *One cut serves three uses. As a level of description it fixes what is modeled and what is summarized; as a control interface it is where a setpoint is imposed and met; as a partition for inference it fixes the cost. The physical hierarchy is a good hierarchy for all three because its interfaces are narrow by design.*

17.6 Worked example: a metered zone under a junction

The two-district network of Chapter 15 had a demand of ten kilograms a second at junction 5. Now junction 5 is the connection point of a district metered area, and that demand is a radial service network of eight nodes with uncertain demands whose prior means sum to ten and whose spreads are a quarter of their means; Figure 17.4 draws it. The zone is a region with twenty-three interior variables, eight heads, seven interior pipe flows and eight demands, and two ports, the inlet flow m_{in} from junction 5 into the zone and the head h_5 on the distribution side. Distribution has its own factors, the balances and closures of Chapter 15 with its three meters, and reads the zone only through the inlet. Every number is from `ch17_zone.py`.

The message is a draw. Eliminating the zone's interior onto its two ports gives a precision of 1.157 on the inlet flow and 5×10^{-12} on the head. The zone says nothing about h_5: with only demands inside and no source of its own, its net draw is the sum of its demands whatever the head at which it is supplied, by Proposition 17.1. Its message is the one-dimensional statement $m_{\text{in}} \sim \mathcal{N}(10.000,\, 0.930^2)$ kg/s, an equivalent draw with error bars, the spread being the demands' prior spreads in quadrature, 0.9294, widened by four ten-thousandths by the physics rows. This is what a water planner's "zone demand at junction 5" has always been, made exact and given a variance. The whole two-level model has 41 unknowns and 44 factors and converges in eight Gauss–Newton iterations; with distribution's three meters it puts the inlet at 10.226 ± 0.679 and the pipe from junction 4 at 8.741 ± 0.667 kg/s.

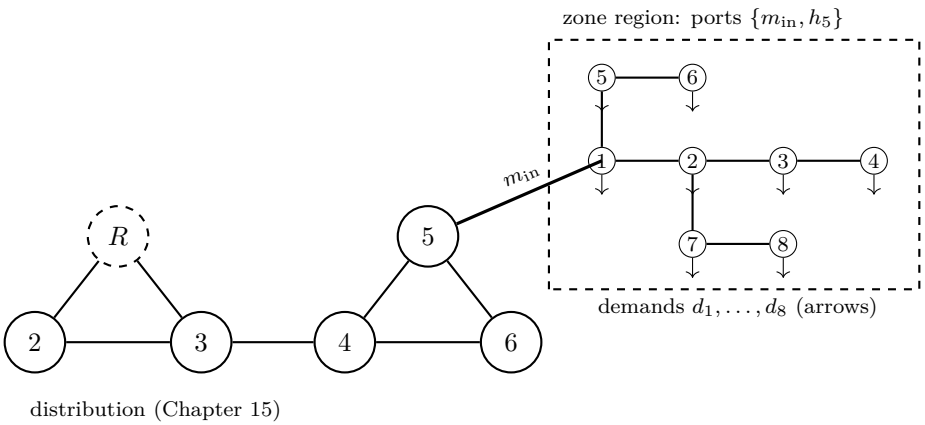


Figure 17.4: A district metered area behind junction 5 of the two-district network. To distribution the zone is a region behind its inlet; its message is a Gaussian on the inlet flow and the head at the connection point.

That the zone says nothing about h_5 is worth a sentence more here than it needed in a domain where potentials are relative anyway. A water network has an absolute potential: the head at a junction is a pressure a gauge could read, and the zone certainly has an opinion about the heads *inside* itself. What it has no opinion about is the head at which it is connected, because nothing inside it references the reservoir. The message is an equivalent draw and not an equivalent resistance, and it would become the latter the moment the zone acquired a second inlet.

A meter at the inlet narrows the interior. Put a meter on the inlet, reading 10.80 ± 0.10 kg/s — which is what a district metered area is for. The inlet's posterior is 10.788 ± 0.099, and the correction passes down: every zone demand moves up from its prior, by 0.16 to 0.46 of its spread, in proportion to its prior variance, which is the virtual-sensor arithmetic of Chapter 12 applied through a port. Each demand's spread shrinks only to between 0.85 and 0.98 of its prior, because one number cannot narrow eight; what the meter has narrowed is their sum, from 0.929 to 0.103 kg/s, and their posterior is correlated to say so. A utility that reads only the inlet knows its zone's total to a tenth of a kilogram a second and each customer's draw hardly better than before, which is exactly the complaint district metering attracts and exactly what the arithmetic predicts.

Failure sets enumerated inside, combined at the port. Chapter 15 enumerated failure sets over a whole system and promised that a hierarchy would enumerate them by region. Here is the mechanism. Each of the zone's eight pipes is a candidate outage; in a radial zone an outage isolates everything downstream, whose demand is lost. Table 17.4 lists the eight, each as a zone message: a draw reduced by the lost demand, with a variance reduced by the lost demands' priors. Distribution's consequence of each is one solve of its reduced system of seventeen variables with the zone's message swapped. Combined with distribution's own seven pipe outages, the fifty-six pairs cost eight zone messages and fifty-six reduced solves, not fifty-six solves of the whole forty-one-variable model. With more zones the saving compounds: a system of m zones with b pipes each has mb zone messages and one reduced system, and the pairs are combinations

Table 17.4: The zone's eight single-pipe outages as messages to distribution, and distribution's flows on pipes 4–5 and 5–6 with each message in place. All flows in kg/s.

pipe out	demand lost	message m_{in}	m_{45}	m_{56}
the inlet itself	10.00	0.000 ± 0.010	-0.791	-0.794
1–2	5.40	4.600 ± 0.688	5.275	-1.249
1–5	2.60	7.400 ± 0.801	7.229	-1.388
2–7	1.90	8.100 ± 0.862	7.705	-1.419
2–3	2.00	8.000 ± 0.857	7.645	-1.415
5–6	1.00	9.000 ± 0.896	8.213	-1.452
3–4	0.80	9.200 ± 0.908	8.325	-1.459
7–8	0.70	9.300 ± 0.913	8.379	-1.462

of held messages.

The port as a setpoint. Distribution, wanting to hold pressure elsewhere, asks the zone to hold its inlet at 8.5 kg/s. In the model that is one factor on the port, a setpoint with a negligible width, which Chapter 13 would call imposing an input, and which in a water network is what a pressure-reducing valve at the zone inlet physically does. The zone's posterior distributes the required reduction of 1.50 kg/s over its eight demands in proportion to their prior variances, shares of 0.289, 0.163, 0.104, 0.046, 0.185, 0.072, 0.104 and 0.035 against prior-variance shares of the same eight numbers to three decimals. That is the zone's own decision to make, on its own model, and with its own priors replaced by the cost of each customer's shortfall it would be an optimization inside the zone; distribution sees only the port, where the flows on pipes 4–5 and 5–6 become 7.120 and -1.380. The parent imposed, the child met, and each used only its own factors and the message between them.

17.7 Hierarchy in time

Chapter 16 unrolled the graph in time and found the storage states to be separators between past and future. A hierarchy in time therefore has two families of separators, the ports of each cut at each step and the storage states of the whole system between steps, and the question is whether they can be used together. They can, in one order, and not in another that is tempting.

Proposition 17.3 (Nested elimination in space and time)**.** *In the unrolled graph of a hierarchy of regions, eliminating at each step the regions' interiors onto their ports and the storage states, and then eliminating the step onto its storage states, is exact. The message carried from step to step is a Gaussian on all the storage states of the system jointly.*

Proof. Within a step, the regions' factors read their interiors, their ports and their own storage states at the previous step; the ports separate the interiors as in Proposition 5.3, so eliminating each interior onto its ports and storage states is a region message, and Proposition 17.2 lets it be done level by level. Across steps, Proposition 16.2 says the storage states separate; eliminating everything at the step except the states gives a Gaussian on the states, which is the filter's carried message. Every elimination is a Schur complement and every one is exact. □

The tempting order is to let each region carry its own storage states and nothing else: a filter per sub-zone, a filter for the zone, each updating from its own readings and its neighbors' ports.

Table 17.5: Filtering the supply zone's twenty demands over eighty steps of thirty seconds, with every state carried jointly (exact) and with each sub-zone carrying only its own demand node and demand and the zone only its two junctions (regional). Errors over the last sixty steps.

	exact	regional
rms error of a demand [kg/s]	1.053	1.056
reported spread of a demand [kg/s]	1.039	0.341
ratio, error to reported	1.01	3.10
fraction of errors beyond two spreads	0.013	0.401
reported spread of the twenty demands' sum [kg/s]	0.948	1.530

The proposition says what that loses. The carried message is a Gaussian on the states *jointly*, and after one step of conditioning on shared readings, a district meter or a trunk-junction logger, the states of different regions are correlated. A district meter that reads five demands at once makes their errors negatively correlated, because a reading that says the total is high says that if one demand is higher than estimated another is lower. A per-region filter keeps each region's block of the covariance and drops the rest, which is to say it treats the regions as independent at the start of every step, and then conditions on the same shared reading again as if it were new information about each.

The mechanism is visible in two sources. Let q_1 and q_2 have independent priors of variance v, and let a meter read their sum, $y = q_1 + q_2 + \epsilon$ with $\epsilon \sim \mathcal{N}(0, \sigma^2)$, once per step, with the sources not moving. Exactly: the sum $s = q_1 + q_2$ has prior variance $2v$ and after n readings has variance $(1/2v + n/\sigma^2)^{-1}$, while the difference $d = q_1 - q_2$ is never read and keeps its variance $2v$; since $q_1 = (s + d)/2$,

$$\operatorname{Var}(q_1 \mid y_1, \dots, y_n) \;=\; \tfrac{1}{4}\Big[\big(\tfrac{1}{2v} + \tfrac{n}{\sigma^2}\big)^{-1} + 2v\Big] \;\longrightarrow\; \tfrac{v}{2} \quad (n \to \infty). \tag{17.5}$$

Half the prior variance is all a sum meter can ever remove from one source. A filter that carries q_1 and q_2 in separate blocks conditions each reading on the current variances v_n of both as if independent, $v_{n+1} = v_n - v_n^2/(2v_n + \sigma^2) = v_n\,(v_n + \sigma^2)/(2v_n + \sigma^2)$, which decreases at every step and tends to zero. It comes to believe each source is known exactly, when only their sum is. Every shared reading is treated as fresh information about each part because the correlation that records "this was already used" has been thrown away.

Table 17.5 measures the loss on the supply zone with storage at the demand nodes and the zone junctions, the demands random-walking by 0.15 kilograms a second per step, and the same readings every step as before. The exact filter's errors are what it reports: root-mean-square 1.05 against a reported 1.04, and one percent of errors beyond two spreads, no more than a Gaussian should have. The regional filters have the same errors and report a third of them. Forty percent of their errors lie beyond two reported spreads. And the error is not one-sided: the exact posterior correlation between two demands in the same district is -0.18, which makes their sum better known than the demands individually, 0.95 against $\sqrt{20} \times 1.04$; the regional filters, having dropped the correlation, report the sum at 1.53, worse than the truth, while reporting each source better than the truth. They are overconfident about the parts and underconfident about the whole, which is the signature of dropped negative correlations. The remedy in the filtering literature is either to carry the joint, as the proposition prescribes, or to fuse with a rule that is consistent for any correlation at the price of conservatism, which is covariance intersection [36]; what is not a remedy is to pretend the regions are independent.

Principle 17.4. *In time, ports separate regions within a step and storage states separate steps, and both eliminations are exact in that order. What is carried between steps is the joint Gaussian on all storage states. A region that carries only its own states forgets the correlations that shared readings created, and reports its parts too well and its whole too badly.*

17.8 In practice

Cut where the engineers cut. The physical hierarchy of a system, its zones, districts and sub-zones or its trunk mains and service reservoirs, was drawn by people who wanted narrow interfaces. Use it. A cut that follows a drawing boundary has ports that already have names and usually have meters.

Put the meter on the ports. A meter that reads a region's total is a factor on the region's port flows by Proposition 17.1. Written that way it belongs to the parent, and the parent can use it before the child's message is available, or with a child that is not modeled at all.

Hold the messages. The value of the hierarchy is reuse, and reuse requires that every region's message and factorization be kept between questions. A change recomputes one path from the changed region to the root; a question about an interior back-substitutes down one path. Nothing else is touched.

Do not lump what you can reduce. A region summarized to one node with one potential is exact for its total flow and wrong for its distribution among the ports. The message costs one elimination and keeps every port. If a lumped model must be used, run its misfit test; the lumped room's misfit was thirty times its degrees of freedom.

Impose at the port, meet inside. A setpoint is a factor on a port. The parent imposes it on the child's message; the child back-substitutes it into its interior and decides, with its own model and costs, how to meet it. Neither needs the other's variables.

In time, carry the joint. A hierarchy of filters that each carry their own states is not a filter; it is a set of filters that condition on shared readings twice. Carry the joint Gaussian on the storage states, or fuse conservatively, and check the reported spreads against the innovations.

17.9 What is missing

The hierarchy of this chapter was given, drawn from the physical layout, and every region's ports were few by construction. Two questions remain. When the layout does not give a hierarchy, or gives one whose regions have many ports, how should a large graph be cut, and what does the cut cost? Chapter 18 answers with the boundary dimension of a partition and measures it on test networks. And when the messages are too many or the regions too large for exact elimination, what approximations respect the hierarchy? Chapter 18 also treats co-simulation, in which regions exchange port values by iteration rather than by message, and shows when it diverges.

Notes and further reading

Aggregation of flows across a cut, Proposition 17.1, is the summation of balance rows and appears wherever conservation laws are discretized; the chapter's contribution is only to say that it places total-reading meters on the ports. The composition of Schur complements, Proposition 17.2, is the quotient formula for Schur complements, and its use to eliminate nested substructures is the substructuring of structural analysis [30] and the nested dissection of sparse direct methods [25]; its extension to dynamics as component mode synthesis is Craig and Bampton's [15]. Kron reduction as the electrical form of the same elimination is [18, 46], and the equivalent injection of Section 17.6 is the Ward equivalent [97] with a variance. The view of the hierarchy as a junction tree is Chapter 8's, following [45, 50]. Model reduction in the control-theoretic sense, which approximates where the message is exact, is surveyed by Antoulas [2]. Hierarchical control as coordination at interfaces is Mesarović, Macko and Takahara's [60]. The failure of per-region filters and the covariance-intersection remedy are Julier and Uhlmann's [36]. The reading of the physical hierarchy as simultaneously a level of description, a control interface and an inference partition is the book's.

Summary

- Summing a region's balance rows gives the balance of the region as a node, exactly; a meter on a region's total is a factor on its ports. Intensive states do not aggregate.
- Schur complements compose, so a hierarchy of regions is a junction tree whose messages are computed level by level. Each level pays for its own variables: the sub-zone six, the district twelve, the zone nineteen, for a model of 187.
- Information enters at a level and flows through the ports. A district meter pins the district's total and leaves each demand wide; only a sub-zone's own logger narrows its demand.
- The parent's model of a child is the child's message, an equivalent network with error bars. A lumped node is exact for the total flow and wrong for its distribution: the lumped zone misplaced 51 kilograms a second and reported a spread of 0.4.
- A local change recomputes one path from leaf to root, which on this machine is fifty times faster at 26 000 variables — but it is not cheaper in arithmetic, because the root solve grows with the number of districts. The saving is a constant, and it argues for cutting the root too.
- The same cut is a level of description, a control interface and an inference partition. A setpoint at a port is imposed by the parent and met inside the child.
- A metered zone's message to distribution is a draw with a variance; its failure sets are enumerated inside and combined at the port; a substation meter narrows its loads through the port.
- In time, carry the joint Gaussian on all storage states. Per-region filters report each source three times too well and the sum too badly.

Exercises

Exercise 17.1. Prove equation (17.3) by block elimination. Write $\mathbf{\Lambda}$ in the blocks (I_1, I_2, S), eliminate I_1 to obtain a 2×2 block matrix on (I_2, S), eliminate its I_2 block, and show that the result equals $\mathbf{\Lambda}_{SS} - \mathbf{\Lambda}_{SI}\mathbf{\Lambda}_{II}^{-1}\mathbf{\Lambda}_{IS}$ with $I = I_1 \cup I_2$, using the block inverse of $\mathbf{\Lambda}_{II}$.

Exercise 17.2. District 1 of the supply zone has seventeen balance rows, two for its junctions and three per sub-zone, each with physics width σ_p. Show that the identity $f_s + f_{ai} - f_p - f_{ao} = \sum_j q_{1j}$ holds in the model with a width of $\sqrt{17}\,\sigma_p$, and hence that placing the district meter on the port flows or on the demands gives posteriors that differ by at most that order. Compare with the 0.9 grammes a second the script found for $\sigma_p = 10$ grammes a second and a meter width of 400.

Exercise 17.3. Lump each district of the supply zone to two nodes instead of one, an inlet node carrying the reservoir and ring-in edges and an outlet node carrying the trunk and ring-out edges, with the district's demand at the outlet node. Recompute Table 17.3. Which entries does the two-node lump fix, which does it still get wrong, and what would the lump need in order to be exact on all six ports?

Exercise 17.4. In the zone example, give each demand a curtailment cost c_k per kilogram a second shed, and replace the Gaussian prior on the loads by the constraint that the shed amounts are nonnegative and the objective $\sum_k c_k s_k$. With the tie held at 0.85, which loads are curtailed? Show that distribution's view of the zone is unchanged by the replacement: the message on the port is still a point at 0.85, and the flows on links 4–5 and 5–6 are the same as in Section 17.6.

Solutions to selected exercises

Exercise 17.1. Write

$$\mathbf{\Lambda} = \begin{pmatrix} A & B & C \\ B^{\mathsf{T}} & D & E \\ C^{\mathsf{T}} & E^{\mathsf{T}} & G \end{pmatrix}$$

on (I_1, I_2, S). Eliminating I_1 gives, on (I_2, S),

$$\begin{pmatrix} D - B^{\mathsf{T}}A^{-1}B & E - B^{\mathsf{T}}A^{-1}C \\ E^{\mathsf{T}} - C^{\mathsf{T}}A^{-1}B & G - C^{\mathsf{T}}A^{-1}C \end{pmatrix} =: \begin{pmatrix} D' & E' \\ E'^{\mathsf{T}} & G' \end{pmatrix},$$

and eliminating I_2 from that gives $G' - E'^{\mathsf{T}}D'^{-1}E'$. On the other side, $\mathbf{\Lambda}_{II} = \left(\begin{smallmatrix} A & B \\ B^{\mathsf{T}} & D \end{smallmatrix}\right)$ has the block inverse, with $D' = D - B^{\mathsf{T}}A^{-1}B$,

$$\mathbf{\Lambda}_{II}^{-1} = \begin{pmatrix} A^{-1} + A^{-1}BD'^{-1}B^{\mathsf{T}}A^{-1} & -A^{-1}BD'^{-1} \\ -D'^{-1}B^{\mathsf{T}}A^{-1} & D'^{-1} \end{pmatrix},$$

and $\mathbf{\Lambda}_{SI}\mathbf{\Lambda}_{II}^{-1}\mathbf{\Lambda}_{IS}$ with $\mathbf{\Lambda}_{IS} = (C;\, E)$ expands to

$$C^{\mathsf{T}}A^{-1}C + C^{\mathsf{T}}A^{-1}BD'^{-1}B^{\mathsf{T}}A^{-1}C - C^{\mathsf{T}}A^{-1}BD'^{-1}E - E^{\mathsf{T}}D'^{-1}B^{\mathsf{T}}A^{-1}C + E^{\mathsf{T}}D'^{-1}E$$
$$= C^{\mathsf{T}}A^{-1}C + E'^{\mathsf{T}}D'^{-1}E',$$

since $E' = E - B^{\mathsf{T}}A^{-1}C$. Hence $G - \mathbf{\Lambda}_{SI}\mathbf{\Lambda}_{II}^{-1}\mathbf{\Lambda}_{IS} = G' - E'^{\mathsf{T}}D'^{-1}E'$, which is the two-stage result. The information vector follows by the same substitution with $\boldsymbol{\eta}$ in place of the last block column.

Exercise 17.2. Each balance row of the row's region is $r_n = \sum_{e \ni n} \pm F_e - q_n$ with $r_n \sim \mathcal{N}(0, \sigma_p^2)$ independently. Summing the seventeen rows, every internal edge cancels and the cut edges remain: $\sum_n r_n = (f_p + f_{ao} - f_s - f_{ai}) - \sum_j q_{1j}$, and the sum of seventeen independent widths σ_p has width $\sqrt{17}\,\sigma_p$. So the model holds the identity to $\sqrt{17} \times 10 = 41$ kg/s. A meter of width 400 kg/s placed on either side of the identity is the same factor up to a width of 41 kg/s added in quadrature, $\sqrt{400^2 + 41^2} - 400 = 2$ kg/s, and the posteriors differ by a quantity of that order in the meter's direction, scaled by how much the meter moves the estimate; the script's 1.6 kg/s is consistent.

Chapter 18

Partitioning at scale

Chapters 8 and 17 took the cut as given. The supply zone came with districts and sub-zones, the network with a district behind a trunk junction, and the ports were few because the engineers who drew the system had made them few. This chapter asks what happens when the cut must be chosen, and what a bad choice costs. The answer has a single currency. Every route through a partitioned model, exact elimination, messages between regions, iteration between regional solvers, enumeration of failure sets region by region, pays in the number of ports: the *boundary dimension* of the partition. A cut with few ports relative to its interiors is cheap on every route; a cut with many ports is expensive on every route and, for the iterative routes, fails.

The chapter measures rather than argues. Three transmission networks of the standard kind, with 73, 118 and 1 354 nodes, and the supply zone of Chapter 17 at four sizes, supply the graphs. On each the chapter measures the elimination width that Chapter 5 promised, the boundary dimension of partitions chosen three ways, the convergence of region-level message passing that Chapter 9 promised, and the convergence of co-simulation. It then reads, from the case studies of Part VI, what the same partitions do when the regions are not Gaussian but sets of failure states, which is the question Chapters 10 and 15 left open. The chapter's own computations, of width, boundary dimension, propagation and co-simulation, are from `ch18_partitioning.py`. The separator compression, the region trees and the data-hall chain are read from the committed results files of the case studies that measured them.

18.1 The cost of a cut

Let a partition divide the variables into region interiors $I_1, \dots, I_R$ and a port set S, as in equation (8.4). Region r has n_r interior variables and k_r ports, and $|S| \le \sum_r k_r$, with equality when no port is shared. The exact route of Chapter 8 costs one factorization of each $\mathbf{\Lambda}_{II}^{(r)}$, k_r solves against it to form the message, and one dense factorization of the $|S| \times |S|$ port system. Write w_r for the elimination width of region r's interior on its own, the largest front its factorization creates.

Proposition 18.1 (Width of the partition ordering)**.** *Eliminating every region's interior before any port, each interior in an ordering of width w_r, produces fronts no larger than $\max_r(w_r + k_r)$ during the interior phase and no larger than $|S|$ during the port phase. The elimination width of the whole is at most $\max\big(\max_r(w_r + k_r),\ |S| - 1\big)$.*

Proof. While region r's interior is being eliminated, no other region's interior has been touched

and none is adjacent, by the bordered structure; the only variables outside I_r that an interior variable of r can be joined to are r's own ports. So a front during that phase is a front of r's own elimination, of size at most w_r, together with at most k_r ports. Once every interior is gone the remaining graph is on S, and any ordering of it has fronts of size at most $|S|$. □

In operations the same accounting reads

$$\text{cost} \;\approx\; \sum_{r} \big(n_r w_r^2 + k_r \, n_r w_r \big) \;+\; \tfrac{1}{3}|S|^3, \tag{18.1}$$

the first term for factorizing each interior with fronts of width w_r, the second for the k_r solves that form the message, the third for the dense port system; back-substitution adds a solve per region, which is lower order. The two region terms are linear in the region sizes, so a partition can be as fine as one likes without penalty as long as the ports do not grow; the port term is cubic, and it is the one a bad cut inflates.

The proposition says that a partition's cost is governed by two numbers: the width of the worst region plus its ports, and the size of the port set. The first is small when regions are small or themselves well structured; the second is the boundary dimension. Nested dissection, the ordering that sparse direct solvers use, is this proposition applied recursively [25]: cut the graph by a small separator, order the two sides first and the separator last, and recurse into the sides. Planar graphs have separators of size $O(\sqrt{n})$ [53], which bounds the elimination width of nested dissection on them by $O(\sqrt{n})$, and physical networks are nearly planar. What the bound leaves open is the constant, and the constant decides whether a network of a thousand nodes has a port system of thirty or of three hundred. The next section measures it.

18.2 Elimination width of physical networks

Table 18.1 gives, for each graph, the elimination width found by the two standard greedy orderings, minimum degree and minimum fill, which are upper bounds on the treewidth of Definition 5.2. The three networks measured here are transmission systems rather than water ones, and deliberately so: the quantity in the table is a property of a graph, not of a domain, and these three are the standard graphs whose widths the sparse-matrix literature has measured for thirty years. They are the IEEE Reliability Test System of 73 nodes in three areas, the IEEE 118-node system, and the 1 354-node PEGASE model of part of the European grid, all taken as their nodal graphs with parallel branches merged. A distribution network of the same size behaves the same way, for the same reason — it is laid out in streets, which makes it nearly planar — and Chapter 21 measures it directly on the EPA's Net3: three candidate interfaces on a 92-junction network, chosen by its articulation points, with the interface's marginal covariance computed both as an inverse Schur complement and as a block of the full inverse and the two differenced. The supply zones are the graph of Chapter 17's zone, its trunk and ring junctions and its districts of sub-zones, at four sizes.

Two things stand out in Figure 18.1. The transmission networks have widths of 4, 5 and 12, a third of $\sqrt{n}$ and less, so that the largest dense block a direct factorization ever forms on the 1 354-bus network has thirteen rows. Exact inference on these graphs is not merely affordable; it is cheap, and the reason is the one Chapter 5 gave, that a transmission network is a sparse, nearly planar graph whose cycles are local. And the supply zone's width is 2 at every size. Its graph is a set of chains (inlet, demand node, outlet) hanging between the two junctions of each district, with every district hanging on the same two zone junctions; eliminate the chains, then

Table 18.1: Elimination width of the nodal graphs, by minimum-degree and minimum-fill orderings, against $\sqrt{n}$. The largest dense block a sparse factorization forms is one more than the width.

graph	nodes	edges	width (min. degree)	width (min. fill)	$\sqrt{n}$
RTS-GMLC	73	108	5	5	8.5
IEEE 118	118	179	4	4	10.9
PEGASE 1354	1 354	1 710	14	12	36.8
zone, 4×5	70	92	2	2	8.4
zone, 8×10	258	344	2	2	16.1
zone, 16×20	994	1 328	2	2	31.5
zone, 32×25	2 466	3 296	2	2	49.7

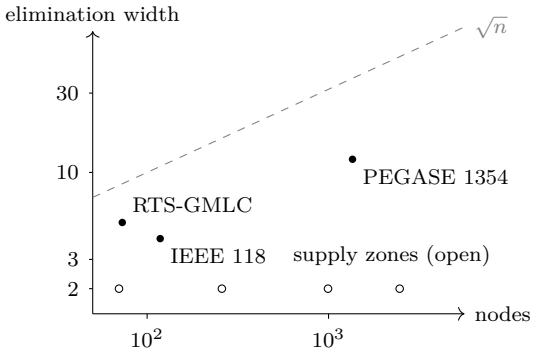


Figure 18.1: Elimination width against the number of nodes, both axes logarithmic. Filled: the three transmission networks. Open: the supply zone at four sizes. The dashed line is $\sqrt{n}$, the planar-separator bound's scaling. The transmission networks sit well below it and the zones do not grow at all.

the district junctions, and the front never exceeds three. This is the hierarchy of Chapter 17 seen by the elimination algorithm: a system built from subsystems with narrow interfaces has an elimination width set by the interfaces, not by the size.

Principle 18.1. *Physical networks have elimination widths far below the planar bound: four to twelve on standard transmission systems of* 73 *to* 1 354 *buses, two on a supply zone at any size. Exact Gaussian inference on them is cheap, and a partition is chosen for reuse and for what it enables, not to make the solve possible.*

18.3 Choosing the cut

Three ways to cut a graph into regions are compared. The *physical* cut follows labels the system carries: the three areas of the Reliability Test System, which are its three synchronous regions. The *spectral* cut bisects the graph by the sign of the Fiedler vector, the eigenvector of the susceptance-weighted Laplacian belonging to its second-smallest eigenvalue [24], splitting at the median so that the halves are equal, and recurses to give $2, 4, 8, \ldots$ regions [74]; a Kernighan–Lin pass [44] could refine it and, on these graphs, changes little. The *random* cut assigns buses to

Table 18.2: Boundary dimension of partitions of the three networks and of two supply zones. Ports are boundary nodes, counted once each; $k_{\max}$ is the largest region's port count. The zones' physical cut is by districts, with the trunk and ring junctions as a root region.

network	cut	R	ports $\lvert S\rvert$	share of buses	$k_{\max}$	cut branches
RTS-GMLC	areas	3	10	14%	4	5
	spectral	2	11	15%	6	7
	spectral	4	26	36%	8	19
	spectral	8	42	58%	9	32
	random	2	60	82%	30	56
	random	8	71	97%	10	106
IEEE 118	spectral	2	16	14%	8	12
	spectral	4	38	32%	12	26
	spectral	8	55	47%	10	39
	random	2	96	81%	50	89
	random	8	118	100%	15	162
PEGASE 1354	spectral	2	30	2%	17	24
	spectral	4	87	6%	29	82
	spectral	8	240	18%	67	211
	spectral	16	359	27%	41	328
	spectral	32	489	36%	27	487
	random	2	950	70%	477	965
	random	32	1 334	99%	43	1 922
zone 8×10	districts	9	18	7%	2	24
	spectral	8	160	62%	32	135
	random	8	256	99%	33	308
zone 32×25	districts	33	66	3%	2	96
	spectral	32	1 532	62%	77	1 416
	random	32	2 464	100%	78	3 194

equal-sized regions at random, and stands for a partition chosen without regard to the physics: by owner, by alphabetical order, by whatever data happened to be at hand.

Table 18.2 and Figure 18.2 give the result. Cut in two, the 1 354-bus network has thirty boundary buses, two percent of the total; cut in two at random it has 950, seventy percent. The ratio is thirty, and it is the ratio of the port systems' sizes; squared, of what the exact route must hold, which is the figure below; and cubed, of the flops of their factorization. The same holds at every R: the random cut's boundary is essentially every bus once R exceeds a handful, because a random region of size n/R in a graph of average degree three has almost no interior. The spectral cut's boundary grows with R, as it must, since more regions mean more cuts, but from a base two orders of magnitude lower. On the Reliability Test System the physical cut, three areas with ten boundary buses, is as good as the best spectral cut and better than the spectral cut into four, which is the observation of Chapter 17 made quantitative: the areas were drawn by engineers who wanted few ties, and on a transmission network the eigenvector finds cuts of the same quality.

The supply zone, cut three ways. The zone makes the same comparison on a hierarchy its engineers drew themselves, and adds a caution. Cut by districts, with the trunk and ring

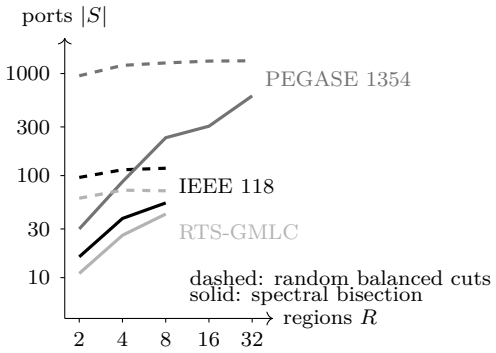


Figure 18.2: Boundary dimension against the number of regions for the spectral cuts (solid) and the random cuts (dashed) on the three networks, both axes logarithmic. A random cut puts most buses on the boundary at any R; a spectral cut's boundary grows roughly as $R^{0.8}$ from a small base.

junctions as a root region of their own, the zone of eight districts and ten sub-zones has eighteen boundary nodes among 258, seven percent, and no region has more than two ports; at thirty-two districts of twenty-five sub-zones the shares are 66 of 2 466, under three percent, and still two ports per region. Cut spectrally into the same number of regions it has 160 boundary nodes, sixty-two percent; cut at random, 256, all but two. The spectral cut fails here for a reason worth knowing: recursive bisection at the median insists on equal halves, and a graph in which every district hangs on the same two hub junctions has no balanced cut that respects the districts, so the Fiedler vector slices across them. The physical cut is unbalanced, a root region of two junctions and thirty-two regions of seventy-seven, and that is exactly what makes it good. Balance is a constraint of parallel computing, not of inference; the cost model of Section 18.1 charges for ports and region widths, and an unbalanced partition with narrow interfaces beats a balanced one with wide ones.

The largest region port count $k_{\max}$ tells the same story from the region's side. A spectral region of the PEGASE network has 17 to 67 ports; a random region of the same size has hundreds. Since a region's message is a dense $k_r \times k_r$ block, and since the discrete enumeration of Section 18.5 is exponential in what the ports can express, $k_{\max}$ is the number that decides whether a region can be summarized at all.

Principle 18.2. *Cut where the flows are thin. A cut chosen by the physics, by the system's own areas or by the Fiedler vector of its weighted Laplacian, has a boundary of a few percent of the variables; a cut chosen without the physics has a boundary of most of them, and every route through the partition pays the difference.*

18.4 Coupling the regions: messages, propagation, or iteration

Once the regions are cut, their interiors are eliminated onto the ports and what remains is the port system, a precision on S built from the region messages. There are three ways to solve it; Figure 18.3 draws the first and the third for two regions.

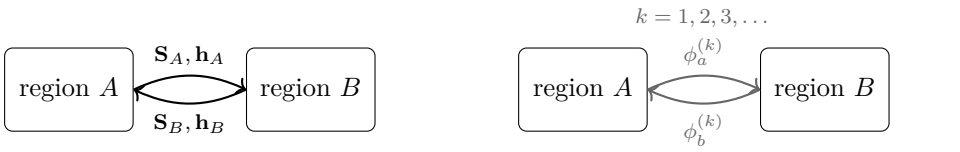


Figure 18.3: Two ways to couple two regions. Left: each region sends the other its Schur complement onto the ports, a precision and an information vector; the port posterior is exact after one exchange. Right: each region re-solves with the other's last boundary potential and sends the new one; the sequence converges to the same answer at a rate set by the tie strength.

The first is exact: factor the port system as one dense matrix. It costs $|S|^3/3$ and never fails. When the region graph is a tree, as for two regions, the factorization is one message each way (Chapter 8).

The second is to pass messages between regions and iterate: Gaussian belief propagation on the region graph, with each region's ports as one vector-valued variable. Chapter 9 found that propagation on the physical graph itself never converges, and asked whether it converges on the region graph, whose factors are messages that have absorbed the tight ties. The walk-summability criterion of Chapter 9 for scalar variables is that the spectral radius of the absolute partial-correlation matrix, $\rho(|\mathbf{R}|)$, is below one; for vector variables the natural analogue replaces each scalar partial correlation by the spectral norm of the normalized block, $\|\mathbf{D}_r^{-1/2}\mathbf{\Lambda}_{rs}\mathbf{D}_s^{-1/2}\|_2$, with $\mathbf{D}_r$ the region's own port block, and asks whether the spectral radius ρ_B of the matrix of those norms is below one. Both are computed here, and the propagation is also simply run, to 10^{-8} or to failure.

The third is co-simulation: each region solves its own physics with its neighbors' boundary potentials held at the last exchanged values, and the regions exchange again. Exchanging simultaneously is block Jacobi on the nodal system; exchanging in sequence is block Gauss–Seidel. Both converge on these systems, since the nodal matrix with a slack is an M-matrix and block splittings of M-matrices are regular [94], but at a rate set by the partition.

The convergence factor of an exchange. The rate has a closed form for two regions joined by one tie. Eliminate each region's interior onto its boundary bus; region A becomes an equivalent conductance g_a from its boundary potential ϕ_a to the reference together with an equivalent injection, region B likewise with g_b, and the tie has conductance g_t. The port system is

$$\begin{pmatrix} g_a + g_t & -g_t \\ -g_t & g_b + g_t \end{pmatrix} \begin{pmatrix} \phi_a \\ \phi_b \end{pmatrix} = \begin{pmatrix} p_a \\ p_b \end{pmatrix}. \tag{18.2}$$

Simultaneous exchange solves each row with the other's potential from the previous round, $\phi_a^{(k+1)} = (p_a + g_t\phi_b^{(k)})/(g_a + g_t)$ and likewise for b, whose iteration matrix has spectral radius

$$\rho_J \;=\; \frac{g_t}{\sqrt{(g_a + g_t)(g_b + g_t)}}, \qquad \rho_{GS} = \rho_J^2 \tag{18.3}$$

for the sequential variant. The factor is near zero when the tie is weak against the regions' own stiffness and near one when it is strong; a tie as strong as the regions it joins gives $\rho_J = 1/2$ and twenty exchanges for six digits, and a tie ten times stronger gives 0.91 and 150. The message

Table 18.3: Coupling the regions of each partition. $\rho(|\mathbf{R}|)$ is the scalar walk-summability radius of the port system; ρ_B is its block analogue with regions as vector variables; the propagation column says whether region-level belief propagation reached 10^{-8} and in how many sweeps; the last two columns are the block Jacobi and block Gauss–Seidel convergence factors of co-simulation on the physics and the sequential exchanges needed for 10^{-6}.

network	cut	ports	$\rho(\lvert\mathbf{R}\rvert)$	ρ_B	propagation	ρ_J	ρ_{GS}	exchanges
RTS-GMLC	areas 3	10	1.05	1.48	converged, 20	0.734	0.541	22
	spectral 2	12	1.03	0.90	converged, 1	0.906	0.821	70
	spectral 4	32	3.00	1.99	converged, 20	0.975	0.950	270
	spectral 8	53	3.00	2.75	failed	0.979	0.958	322
	random 4	139	2.69	3.00	failed	0.995	0.990	1 364
IEEE 118	spectral 2	19	2.17	1.00	converged, 1	0.872	0.761	51
	spectral 4	43	2.97	1.86	failed	0.968	0.937	213
	spectral 8	64	2.92	2.89	failed	0.975	0.951	273
	random 4	192	2.93	3.00	failed	0.996	0.991	1 560
PEGASE 1354	spectral 2	37	2.99	1.00	converged, 1	0.961	0.924	175
	spectral 4	117	3.00	2.84	failed	0.991	0.983	791
	spectral 8	332	3.00	4.62	failed	0.995	0.990	1 308
	spectral 32	770	3.00	7.20	failed	0.999	0.998	8 843
	random 4	2 211	3.00	3.00	failed	1.000	1.000	55 369

route computes the fixed point of this iteration in one step, because the message *is* g_a, g_b and the injections, and the port solve inverts the 2×2 matrix. With more ports the factor is the spectral radius of the corresponding block iteration, and the closed form becomes a measurement.

Table 18.3 gives the measurements; the model is the linear-Gaussian one of Chapter 6 in the book's own variables, angles, flows and injections, with a physics width of 10^{-3}, injection priors of spread 0.1 and flow meters on a third of the branches. Four things are worth reading off.

First, the scalar walk-summability radius of the port system is above one for every partition, at 1.03 to 3.00, as it was 3.2, 5.1 and 8.1 for the three full graphs. Eliminating the interiors does not make the port system walk-summable in the scalar sense; the region messages are dense blocks, and dense blocks of partial correlations of order one half among a region's own ports are exactly what Chapter 9 found fatal. Scalar propagation among the ports is no better than scalar propagation among the buses.

Second, block propagation, with regions as vector variables, behaves as Chapter 9 predicted. On two regions the region graph is a tree and one sweep is exact. On the Reliability Test System's three areas, whose region graph is a triangle, it converges in twenty sweeps, and on its spectral cut into four, also in twenty; on every partition into eight or more, and on every random partition, it fails, the beliefs' precisions turning indefinite within a sweep. The block radius tracks this: below one on the two-region cuts, 1.5 and 2.0 where propagation converged on a cycle, 2.8 and above where it failed. The condition $\rho_B < 1$ is sufficient, and the measurements say that a small excess over one is survivable on a region graph with one short cycle and a large excess is not. Region-level propagation is therefore a tool for a few large regions, and not a substitute for the port solve when the regions are many.

Third, co-simulation always converges on the physics, and the rate is what equation (18.3) predicts. On the three areas, with five tie lines against three regions of two dozen buses, the

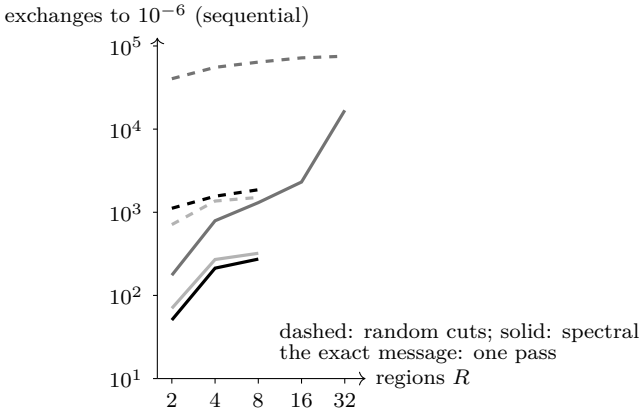


Figure 18.4: Sequential exchanges of co-simulation needed for six digits, against the number of regions, for spectral (solid) and random (dashed) cuts of the three networks. The exact message route needs one. Both axes logarithmic.

Jacobi factor is 0.73, the Gauss–Seidel factor is 0.54, close to the square 0.54 that the two-region formula gives, and six digits take twenty-two exchanges. On two spectral regions of the PEGASE network the factor is 0.92 and six digits take 175 exchanges; on thirty-two regions, 8 843; on a random cut into four, 55 000, the factor being one to four decimals. Figure 18.4 plots the exchange counts against the number of regions for both kinds of cut on the three networks. Each exchange is a full regional solve. The message route does the same regional solves once, plus the port factorization, and is finished.

Fourth, the port count orders every column. Read down any network's rows: as $|S|$ grows the block radius grows, propagation goes from converging to failing, and the exchange count grows by an order of magnitude and more. The boundary dimension is the one number that predicts the cost of every route.

The port count as a cost predictor. The exact route was also counted, in one unit: everything it forms and holds. That is the nonzeros of each region's sparse factor, plus the $|S|^2$ entries of the dense port block it accumulates them into. Figure 18.5 plots that count against the port count for every partition of the three networks, and the points from three networks and two kinds of cut fall on one curve with two parts.

It is *flat* while the regions dominate. On the 73-bus network the spectral cuts into two, four and eight hold 4 525, 4 023 and 4 816 numbers while $|S|$ goes from 12 to 32 to 53: refining the partition shrinks the interiors about as fast as it grows the boundary, and the total does not move. The floor is the network's own sparsity and the partition cannot get below it.

It rises as $|S|^2$ once the port block dominates, and by the largest cuts it is *all* port block. The random cut of the 1 354-bus network into thirty-two regions has $|S| = 2\,785$ and holds 7 760 557 numbers; $|S|^2$ is 7 756 225. Everything else has become rounding. Between the two, the spectral cut of the same network into thirty-two has $|S| = 955$ and holds 946 317 against an $|S|^2$ of 912 025 — the port block is already 96 percent of it.

That is Proposition 18.1 drawn: below the floor the partition cannot help, and above it the cost is a function of the ports and of nothing else about the partition. A cut is worth making until the port block reaches the floor, and after that a cut is only worth making for the reuse.

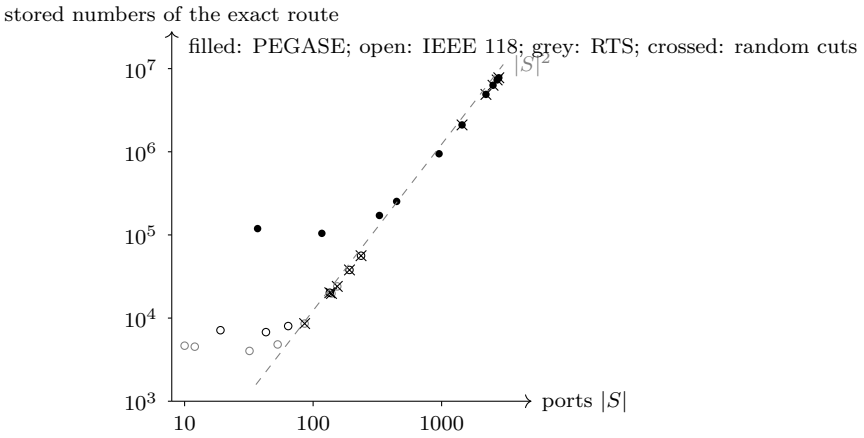


Figure 18.5: What the exact route forms and holds — the nonzeros of every region's sparse factor plus the dense port block — against the port count, for every partition of the three networks, both axes logarithmic. The dashed line has slope two. The points of three networks and two kinds of cut lie on one curve: flat at a floor the regions set, then rising as $|S|^2$ once the port block dominates.

The route was timed as well, and the clock agrees with the count on every ordering that matters: random cuts cost more than spectral ones at the same region count, the 1 354-bus network costs more than the other two, and refining a spectral partition is nearly free until the port block grows. The timings themselves are not quoted here and not committed — they are one machine's, they move between runs of unchanged code, and the script prints them for whoever wants them. What the clock does add is the comparison the count cannot make: a monolithic sparse solve of the whole model is faster than any partition of it. That is Chapter 17's point again. The partition is not for the one solve. It is for the second question, and the tenth.

Principle 18.3. *Between regions, pass the exact message and solve the port system. Scalar propagation on the ports fails as it does on the buses; block propagation converges only among a few large regions; co-simulation converges at a rate set by the ties' strength against the regions' and needs tens to tens of thousands of exchanges where the message needs one. Every route's cost rises with the boundary dimension.*

18.5 Enumeration by region

The Gaussian measurements above bound what a partition costs when the regions' messages are precisions. Chapters 10 and 15 asked a harder question: when a region's state is a set of failed components, its message is not a Gaussian but a distribution over what the ports can take, one value per distinct separator value the region's failure states produce, and the question is how many such values there are. If many failure states of a region produce the same port values, the region compresses: its $|E_r|$ states become $|S_r|$ separator values, and the regions combine on the ports at the cost of $|S_r| \times |S_s|$ pairs rather than $|E_r| \times |E_s|$. If they do not, the partition buys nothing on this route.

A companion set of partitioning studies in the power domain, committed to the repository that accompanies the book, measured the compression ratio $C_r = |E_r|/|S_r|$ on the same three networks,

Table 18.4: Separator compression measured by a case study of Part VI: enumerated region failure states per distinct separator value, for the physical, spectral or refined spectral (spectral+KL) cut of each network, with the boundary dimension of the region measured. Ranges run over the regions of the cut. The last column is the partition the study found cheapest by measurement against the one the port count alone predicted.

network	cut	k_r	C_r	best: measured / predicted
RTS Area 1, two replicas, one tie	replica	1	15.9	–
RTS Area 1, two replicas, two ties	replica	2	3.4	–
RTS-GMLC, three areas	areas	2–4	3.7–5.7	areas / areas
RTS Area 1 (24 buses)	spectral	3–4	1.8–2.5	spectral / spectral
RTS Area 1 (24 buses)	spectral+KL	3–4	1.6–3.3	–
IEEE 118	spectral	8	1.02, 1.63[b]	spectral+KL / spectral
PEGASE 1354	spectral	13–17	1.5–2.0[a]	spectral / spectral

[a]Enumerated to a residual target of 10^{-1} with a cap of 5 000 states per region; the cap binds, with residual mass 0.97 and 1.00, so both ratios are censored.
[b]The second region stopped at the cap of 100 000 states with residual mass 0.032, so its ratio is censored. The spectral cuts of the full RTS-GMLC, not tabulated, stop at the same cap with residuals 0.023 and 0.026.

with separator values distinguished at the precision the physics gives. On the Reliability Test System and the IEEE 118-bus system it took physical, spectral and random cuts and enumerated each region's failure states to a residual mass of 10^{-2}, with a cap of 100 000 states per region. On the PEGASE system it took spectral cuts only, enumerated to a residual target of 10^{-1} with a cap of 5 000 states, and the cap binds: the enumerated states leave a residual mass of 0.97 to 1.00. Where a region stops at its cap, its ratio is censored, measured on a shallower enumeration than the target. Table 18.4 summarizes what the committed results contain, and its notes mark the censored rows.

The finding is sobering and instructive. Compression is large only where the cut is a single tie and the region has interchangeable parts: a replica of the Reliability Test System's Area 1 behind one tie compresses sixteenfold, because its many twin generating units produce the same tie flow whichever twin fails. It falls to three at two ties and to between one and six on the real networks. On the IEEE 118-bus system, whose generating units are all different, it is 1.02 on one region, where every failure state is its own separator value, and 1.63 on the other, measured at the state cap; the partition saves almost nothing on the enumeration route. The study's own gate, a median compression of ten at three ports or fewer on two grids, was not met, and its verdict is recorded as unfavorable. What the partition does predict, even there, is which cut is cheapest: on three of the four grids the partition with the fewest ports was also the one that measured cheapest, and on the fourth the two differed by a refinement pass.

This is the honest answer to Chapter 15's question of how large a region can be before enumeration by region stops working. It works when the region's failure states collapse onto few port values, which is a property of the physics inside the region, twin units, radial feeders, symmetric layouts, and not of the cut alone. Where the regions are heterogeneous, the port values are as many as the states, and the route through the ports is a reorganization of the enumeration rather than a reduction of it. The Gaussian route has no such condition, because a Gaussian message is a fixed-size object whatever the region contains, and this is the strongest practical argument for the Laplace approximation of Chapter 7 within regions wherever it is

Table 18.5: Region trees for chains of R single-tie regions, read from a case study of Part VI: the joint state count, the certified bracket on the probability of any overload before the residual mass is added, and the reference. At $R = 2$ the reference is the exact value on the same enumerated mass. Wall-clock is that study's last refinement pass.

R	components	joint states	bracket	reference	pass time
2	124	$10^{8.1}$	$[0.1207, 0.1227]$	0.1227 (exact)	7 s
3	186	$10^{12.2}$	$[0.1000, 0.1023]$	0.113 (MC)	63 s
4	248	$10^{16.2}$	$[0.0765, 0.0793]$	0.088 (MC)	230 s
6	372	$10^{24.3}$	$[0.0452, 0.0476]$	0.056 (MC)	511 s

valid.

What the discrete message costs. When the regions are discrete the message is a histogram: one entry per distinct separator value, carrying the total probability of the region states that produce it and, for each consequence of interest, their contribution. Combining two regions is then a sum over pairs of entries, $|S_r| \times |S_s|$ of them, and the physics is evaluated once per region state, not once per pair, because a region state's flows are fixed by its own separator value and the other region's. The two-replica system with one tie, dispatched at peak load, shows the arithmetic at a residual of 10^{-2} per region. Each region enumerates 11 336 states, which fall on 712 separator values, so one table over pairs of separator values has $712^2 = 506\,944$ entries; that is the interface count of Table 18.4. The two-region pass builds one such table for each region, 1 013 888 interface pairs in all, with 21 916 physics evaluations. Its bracket on the probability of any overload, $[0.664, 0.684]$, has the joint residual mass 0.020 as its width and contains the Monte Carlo reference 0.679. The global enumeration of the same system, with 100 000 states, reached $[0.58, 0.69]$. It needs five million states to reach a residual of 0.019, about the two-region residual, which the regional route reaches with 22 672 enumerated states, some 220 times fewer. The difference is ordering: the global enumeration must order 124 components' failure sets by probability, while the regional one orders sixty-two of them twice and takes the product. Where the ports compress, the pairs are far fewer than the states; where they do not, the pairs are the states, and only the ordering advantage remains.

Where the condition holds, the same studies show what it buys. Table 18.5 reads their results for chains of Area 1 replicas joined by single ties, whose region graph is a path and whose messages therefore combine exactly on a one-dimensional lattice of tie flows. The joint state count grows from 10^8 for two regions to 10^{24} for six, beyond any enumeration; the region-tree pass brackets the probability of any overload on the enumerated mass to a width of two to three thousandths in minutes. With the residual mass added to its upper end, the bracket contains the Monte Carlo reference at every length; without it, the reference lies above the bracket for three regions or more. These chains are dispatched pro rata at sixty percent load, so their two-region member is not the peak-dispatch system of the previous paragraph: its probability of any overload on the enumerated mass is 0.1227, where the peak system's is near 0.68.

Two remarks on that table. The Monte Carlo references at $R \geq 3$ lie above the bracket, by about one percentage point, 0.011, 0.009 and 0.008; the bracket is on the enumerated mass, and the un-enumerated residual, 10^{-2} per region and hence up to six percent for six regions, is what the reference contains and the bracket does not. The study reports the bracket with the residual added, 0.132, 0.119 and 0.106, which does contain the reference in each case. And the

Table 18.6: The data-hall chain, read from a case study of Part VI: the two-hall system solved by two-region messages against brute force, and the three-hall system bracketed by messages against Monte Carlo. The three-hall bracket includes the residual mass: it runs from the largest single hall's overload probability to the union bound over hall and wall events plus the joint residual.

	two halls, one wall	three halls, two walls per interface
components / joint states	12 / 4 096	28 / $10^{4.7}$
ports per hall	1	2
probability of any overload	0.15773 (messages)	$[0.011, 0.025]$ (bracket)
reference	0.15773 (brute force)	0.01275 (Monte Carlo, 20 000 draws)
agreement, probabilities	4×10^{-16}	reference inside the bracket
agreement, expected severity	8.9×10^{-12} W	–
time	13 ms against 211 ms	0.2 s against 0.66 s

wall-clock grows with R although the per-region work does not, because the lattice on which the messages combine widens with the number of regions; the price of combining exactly is paid on the separator, as everywhere in this chapter.

18.6 A second domain, on purpose

Everything so far has been water or transmission. The same partition machinery, separators, exact two-region messages, certified brackets, was also run in the companion partitioning studies on a chain of data halls modeled as a steady thermal network: racks with heat loads, cooling units with capacity, plenum and wall paths with conductance and a heat-flow limit. The numbers below are that study's, not a water study's relabelled, and they are kept here because the point of the section is what survives a change of domain. The domain entered through a forty-line adapter, conductance for susceptance, heat for power, a cooling unit for a generator, and the probability layer never saw a room. Table 18.6 summarizes the two systems. Two halls joined by one wall path, twelve failing components and 4 096 joint states, gave the probabilities, of any overload 0.15773 and of islanding, equal to brute force to within 4×10^{-16}, and the expected severity, 1 754 W, to within 8.9×10^{-12} W, in 13 milliseconds against 211 for the brute force. Three halls with two wall paths per interface, twenty-eight components and $10^{4.7}$ joint states, gave a bracket of $[0.011, 0.025]$ on the probability of any overload. Its lower end is the largest single hall's overload probability; its upper end is the union bound over hall and wall events with the joint residual mass, 0.003, added. It contains the Monte Carlo reference of 0.01275, 255 overloads in 20 000 draws, and was computed in a fifth of a second against two thirds of a second for twenty thousand draws, and, what a Monte Carlo cannot, a certified per-path bracket that includes the walls themselves, which on this system do overload when a cooling unit is lost and heat is pushed to the neighbors.

That the supply zones of Table 18.1 have elimination width two, and that the data halls compress and combine exactly on their walls, are the same fact, and it is not a fact about water: engineered flow systems of either kind are hierarchies with narrow interfaces, and every route through a partition of them is cheap. The transmission networks are harder on every count, wider, with more ports per region and with heterogeneous regions that do not compress, and they are still far from the bounds. Between the two lie most physical systems.

18.7 In practice

Measure the width before partitioning. A minimum-degree ordering costs nothing and gives the elimination width. If it is small, and on physical networks it is, exact inference on the whole is cheap and a partition is for reuse, for hierarchy and for enumeration, not for tractability.

Take the physical cut when there is one, the spectral cut when there is not. Areas, feeders, buildings and rows are cuts with few ports by design. Without labels, the Fiedler vector of the weighted Laplacian finds cuts of the same quality on a meshed network; on a hub-and-spoke system it does not, because it insists on balance, and the physical cut, unbalanced, is the right one. Never partition by anything that ignores the graph, and never require balance that the physics does not have.

Count the ports and believe them. The boundary dimension predicts the port solve, the message sizes, the convergence of propagation, the exchanges of co-simulation and, where regions compress, the enumeration. A candidate partition with more ports is worse on every route.

Solve the port system; do not iterate on it. Scalar propagation on the ports fails. Block propagation is for a handful of regions. Co-simulation converges but pays a regional solve per exchange, and the number of exchanges is set by the ties against the regions. The message and the port factorization do the work once.

Expect compression only where the region is uniform. A region behind one tie with interchangeable parts compresses its failure states by an order of magnitude; a heterogeneous region does not. Check the ratio on the region before building the enumeration around it, and use a Gaussian message within the region wherever the Laplace approximation holds.

18.8 What is missing

Part V has treated time, hierarchy and scale as the three directions along which a model grows, and found in each the same structure: a separator, a message, a cost set by the separator's size. What the part has not done is run the whole machine on a whole problem from the operator's question to the answer, with every choice of Chapters 1 to 18 made and defended against a naive baseline. That is Part VI, one chapter per study, each read cold, each with its numbers rendered from the committed results that the two tables of this chapter have already drawn on.

Notes and further reading

Elimination width, treewidth and the greedy orderings are Chapter 5's, following [3, 78, 102]; nested dissection is George's [25] and the planar separator theorem Lipton and Tarjan's [53]. Spectral bisection is Fiedler's vector [24] as Pothen, Simon and Liou used it for sparse-matrix orderings [74]; Kernighan–Lin refinement is [44], and multilevel partitioners such as METIS [40] combine the two at scale. Block splittings and their convergence for M-matrices are treated in the domain-decomposition literature [94]; co-simulation as the exchange of interface values between subsystem solvers, and the conditions under which it converges or diverges, is surveyed by Gomes and co-authors [26]. Walk-summability is Malioutov, Johnson and Willsky's [58]; the

block analogue ρ_B is the chapter's own measurement device and is offered as a sufficient condition, not as a theorem. The measurements of elimination width, boundary dimension, region-level propagation and co-simulation on the three networks are the book's; the separator compression, the region trees and the hydraulic chain are read from the case studies of Part VI, which state their own assumptions, in particular that reliability data for the IEEE 118 and PEGASE systems are class rates borrowed from the Reliability Test System.

Two water-network estimators partition for the reasons of §18.1. Han, Eguchi, Mehrotra and Venkatasubramanian [33] cut a large damaged network at its articulation points so that components can be estimated independently under disaster conditions, accepting the coupling through the bridge flows that Chapter 5's Notes describe. Irofti, Romero-Ben, Stoican and Puig [35] cut in time rather than in space: a leak-free estimation over a window, then a localization graph on its residuals, each solved once, which bounds the cost per scenario on their 268-junction benchmark at the price of passing point estimates rather than posteriors between the two stages. The incremental re-elimination of Dellaert and Kaess [17], which re-solves only the part of the Bayes tree that new factors touch, is the robotics form of §18.3's question of where the cut should fall when the graph grows.

Summary

- The elimination width of a partition ordering is at most the worst region's width plus its ports, or the port count; every route through a partition pays in the boundary dimension.
- Measured widths: 5, 4 and 12 on transmission networks of 73, 118 and 1 354 buses, 2 on a supply zone at any size. Exact inference on physical networks is cheap.
- A physical or spectral cut of the 1 354-bus network in two has 30 boundary buses; a random cut has 950. Random regions have almost no interior.
- Scalar propagation on the ports is not walk-summable on any partition. Block propagation converges among two to four large regions and fails beyond. Co-simulation converges at the rate $g_t/\sqrt{(g_a+g_t)(g_b+g_t)}$ and needs 22 to 55 000 exchanges where the message needs one.
- Enumeration by region compresses only where the region is uniform: sixteenfold behind one tie on a system of twin units, hardly at all on the IEEE 118-bus system. Where it compresses, chains of six regions with 10^{24} joint states are bracketed to a few thousandths on the enumerated mass.
- The same machinery on a chain of data halls reproduces brute force to 4×10^{-16} in probability and 8.9×10^{-12} W in expected severity, sixteen times faster.

Exercises

Exercise 18.1. Derive equation (18.3): write the block Jacobi iteration matrix of the two-port system, find its eigenvalues, and show that the block Gauss–Seidel matrix has the square of the Jacobi radius. Then evaluate both for a tie of conductance 10 joining regions of equivalent conductance 2 and 5.

Exercise 18.2. Prove that the elimination width of a graph is at least the size of its largest clique minus one, and use Proposition 18.1 to show that a partition of the supply zone into districts, with the two zone junctions as ports, has an elimination width of at most three whatever the number of districts and sub-zones. Compare with Table 18.1.

Exercise 18.3. Take the IEEE 118-bus network's spectral cut into two and compute the block radius ρ_B of the two-region port system with the regions' own port blocks as normalizers. It is 1.00 in Table 18.3, yet propagation converges in one sweep. Explain why a two-region system converges regardless of ρ_B, and what ρ_B measures there.

Exercise 18.4. A region behind a single tie contains m identical generating units, each failed independently with probability q, and the tie flow depends only on how many are failed. Compute the number of enumerated states with probability above ϵ and the number of distinct separator values, and hence the compression ratio C as a function of m, q and ϵ. Then replace the units by m units of distinct sizes and show that $C = 1$.

Solutions to selected exercises

Exercise 18.1. With D the block diagonal $\mathrm{diag}(g_a+g_t,\ g_b+g_t)$ and $E = A-D$ the off-diagonal part, block Jacobi iterates $\phi^{(k+1)} = D^{-1}(p - E\phi^{(k)})$, whose iteration matrix is

$$M_J = -D^{-1}E = \begin{pmatrix} 0 & \dfrac{g_t}{g_a+g_t} \\ \dfrac{g_t}{g_b+g_t} & 0 \end{pmatrix}, \qquad \lambda^2 = \frac{g_t^2}{(g_a+g_t)(g_b+g_t)},$$

so $\rho_J = g_t/\sqrt{(g_a+g_t)(g_b+g_t)}$. Gauss–Seidel updates a then b with the new ϕ_a: $M_{GS} = -(D+L)^{-1}U$ with L the strictly lower and U the strictly upper part, which for a 2×2 system gives $M_{GS} = \left(\begin{smallmatrix} 0 & \rho_{ab} \\ 0 & \rho_{ab}\rho_{ba} \end{smallmatrix}\right)$ with $\rho_{ab} = g_t/(g_a+g_t)$ and $\rho_{ba} = g_t/(g_b+g_t)$; its nonzero eigenvalue is $\rho_{ab}\rho_{ba} = \rho_J^2$. For $g_t = 10$, $g_a = 2$, $g_b = 5$: $\rho_J = 10/\sqrt{12\times 15} = 0.745$ and $\rho_{GS} = 0.556$, so six digits take $\log 10^{-6}/\log 0.745 = 47$ simultaneous or 24 sequential exchanges.

Exercise 18.2. In any elimination ordering, consider the first vertex of a largest clique Q to be eliminated. At that moment every other vertex of Q is still present and adjacent to it, since edges are never removed, so its front has at least $|Q|-1$ vertices, and the width is at least $|Q|-1$. For the supply zone, take each district as a region with interior its two junctions and its sub-zones' nodes, and the two zone junctions P and A as the port set, $|S| = 2$. Within a district, eliminate each sub-zone's chain inlet-to-outlet first: the inlet node has neighbors C and D, the demand node then has C and O, the outlet then has C and X, so w_r counts at most two interior neighbors, and each district junction is adjacent to at most the two ports, so $w_r + k_r \le 2+2 = 4$ is the proposition's bound; a sharper count notes that a sub-zone node's front never contains both ports, giving 3. Then the two district junctions each see $\{P, A\}$ and each other's eliminated fill, a front of three. The port phase is on two junctions. The width is at most three at any size, and Table 18.1 measures two, one better, because the minimum-degree ordering eliminates the districts' junctions before the ring junction and never forms the three-clique.

Part VI

Case studies

Chapter 19

Case studies

The chapters before this one taught the methods on networks small enough that every number could be checked by hand. The chapters of this part apply them to networks where almost no number can be checked by hand, and report what happened. Each is a complete application of the book to one question on one network, and each was run as a study in its own right, with its code and its results committed to the repository that accompanies the book.

19.1 What a case study is here

A case study in this part is a question, a system, a model and a run. The question is one a water utility engineer would ask. The system is a published benchmark network, or a real distribution network with a year of recorded data, described fully enough to rebuild. The model is a factor graph of the kinds in Part I, with the uncertainty placed where Part II puts it. The run is a script that produces a results file, and every number in a chapter of this part is read from that file by a script in the book's `scripts` directory. Nothing is retyped.

Every study was run with TERRA, the author's implementation of the constructions of this book, and its plugins. The engine compiles a conservation graph into its residual rows, solves them by Newton's method, and lifts the same rows into the MAP and Laplace inference of Part III. The plugins carry what is specific to a domain or a method: the head-loss closures and network factories, the leak-hypothesis machinery, the readers and scoring code for the published benchmarks, and the properties of water. Appendix C describes both and maps each construction of the book to the module that implements it. In the chapters of this part, "the engine" means TERRA. The studies count cost in *engine solves*, one evaluation of the physics at one setting of the uncertain inputs, and in every chapter of this part that is a Newton or MAP solve of a compiled hydraulic model. Unlike some domains, no study here has an inner problem that bypasses the Newton solver. That makes the counts directly comparable across the four chapters, and it is why the screening result of Chapter 22 can be stated as a ratio of solves without qualification.

The chapters share one order. Each states the question, describes the network, draws the model as a factor graph, lists the method with a pointer to the chapter that justifies each step, gives the results, says what surprised, and ends with how to reproduce the run. The section on what surprised is not decoration. Several of these studies found that a method did less than expected, that a published number could not be reproduced as published, or that a result held only because of an assumption the study had not meant to lean on, and those findings are

reported with the same care as the ones that went well.

19.2 Each chapter removes an advantage

The four chapters are ordered by what the modeller is allowed to check against. Each gives up one thing the previous one had, and by the last there is nothing left but the benchmark's own scorer. That order is the point of the part: a method that survives the removal of every crutch in turn is worth more than one demonstrated once under the best conditions.

- Chapter 20, *Three questions on one set of rows*: a reservoir, two pipes in parallel and one in series. State, leak size and leak location are answered from the same rows, and every one of those answers is checked against a closed form *and* against an independent hydraulic solver. This is the only chapter in which both checks are available.
- Chapter 21, *Half the pipes have meters*: the EPA's Net3, two hundred and eleven unknowns, no closed form. The closed form is gone; what remains is a simulator that can produce the truth for any draw, so the posterior's error bars can be tested for coverage rather than assumed.
- Chapter 22, *Screen, then confirm*: a two-hundred-and-sixty-eight-junction benchmark from the leak-detection literature, where ranking every hypothesis exactly is no longer affordable. The per-draw truth is gone; a day believed to be leak-free takes its place, and Section 22.5.4 shows how much of the result was resting on it.
- Chapter 23, *Held to what a utility knows*: a year of recorded operating data from L-Town, scored by the benchmark's own published economic scorer against six teams' entries. There is no closed form, no per-draw truth, and no day known to be clean, because leaks run for months and overlap. What is left is the scorer.

19.3 What is not here

Two things a reader might expect are absent, and it is worth saying why.

There is no study of a network carrying a second conserved quantity. The machinery for it is in the book — Chapter 2 builds the two-domain graph and Example 1.3 works one — and a pressurized network that also carries heat, or chlorine, is the same graph with a second ledger on the same nodes and edges. But the questions that ledger answers are a different engineer's, and they are the subject of a companion edition rather than of this one. Nothing in Chapters 1 to 18 assumes the graph carries only mass.

And there is no study of forward uncertainty at scale on a real water network: no census of solves over uncertain demands across a whole distribution system. Chapters 10 and 14 build that machinery and demonstrate it on a scheme small enough to have a closed form. Putting it on a real network is the most valuable study this part does not yet contain.

19.4 Reading a case study

Each chapter can be read on its own after the chapters it points to. A reader who wants to rerun a study needs the repository, a Python environment with the engine of Appendix C, and the command in the chapter's last section. Most studies run in minutes on a workstation; the chapters say where one takes longer, and the L-Town study of Chapter 23 is the one that does. A

rerun reproduces the committed results file, and the book's script for the chapter then reproduces every number in the chapter from it.

Chapter 20

Three questions on one set of rows

Numbers in this chapter are produced by `scripts/w0_smallest_loop.py` from the study's committed results file. Nothing is retyped.

20.1 The question

A water utility owns a district with two pressure gauges in it. Over a week the gauges drift downward by a few tens of centimeters. Three questions follow, and in most utilities three different tools answer them:

1. *What is the state of the district?* A hydraulic model, calibrated once a year, is run to give heads and flows everywhere.
2. *Is water being lost, and how much?* A water-balance spreadsheet subtracts metered consumption from metered supply and reports the difference.
3. *Where is it being lost?* A crew walks the mains with acoustic equipment, or a step test isolates sections at night.

The three tools do not share a representation. The hydraulic model does not know what the gauges read. The water balance does not know the pipe network. The crew does not know either. Each answer is produced, and then a person reconciles them.

This chapter shows that the three questions are one set of equations asked three ways. The state is the most probable configuration of that set. The leak's size is one more unknown added to it. The leak's location is a choice among the versions of the set that place the unknown at different junctions, decided by which version explains the readings best once the freedom it grants has been paid for. No new physics enters between question one and question three; only the graph changes, and it changes by one node and one row.

The system is the smallest on which that claim can be checked completely. It has a loop, so it is not a tree and its flows are not determined by topology alone. It has a series element, so a fault at one junction propagates to another. It is small enough that its leak-free state has a closed form, and simple enough that an independent reference solver can be given the same network. Both checks are available at once, which is the point: a claim about inference on a city network is worth nothing until the equations underneath it reproduce answers that can be verified without them.

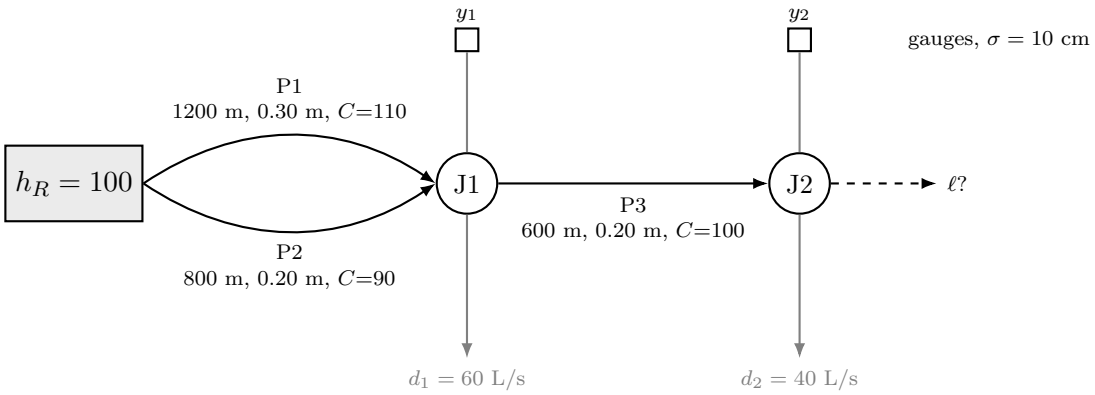


Figure 20.1: The district. A reservoir, two pipes in parallel, one in series, two metered demands, two pressure gauges, and an unmetered outflow ℓ that may or may not be there. The loop makes the flow split a question; the series element makes a fault at J2 visible at J1.

20.2 The system

Figure 20.1 draws it. A reservoir holds a fixed head of 100 m — head is the water's energy per unit weight, measured as a height of water, and Chapter 1 calls it the potential of the hydraulic domain. Two pipes, P1 and P2, run in parallel from the reservoir to junction J1. A third, P3, runs from J1 to J2. J1 delivers 60 L/s and J2 delivers 40 L/s, both metered and treated as known. Pressure gauges sit at J1 and J2 and read head with a standard deviation of 10 cm. A leak, when present, is an unmetered outflow at one of the two junctions.

The head a pipe loses is a fixed, increasing function of the flow it carries. Water distribution conventionally uses the Hazen–Williams law, in which the loss grows as the flow to the power 1.852, with a coefficient set by the pipe's length L_e, diameter D_e and a roughness number C_e that is larger for smoother pipe:

$$K_e \;=\; \frac{10.667\,L_e}{C_e^{1.852}\,D_e^{4.871}}, \qquad \Delta h_e \;=\; K_e\,Q_e^{1.852}, \tag{20.1}$$

with Q_e the volumetric flow in m^3/s and K_e in SI units. For the three pipes of Figure 20.1 this gives $K_1 = 747.32$, $K_2 = 5206.65$ and $K_3 = 3212.75$. Note how strongly diameter enters: P2 is shorter than P1 but two thirds its diameter, and its coefficient is seven times larger.

Equation (20.1) is a closure in the sense of Chapter 1: a relation between a flow and the potentials at its ends, with no preferred direction. It is not an instruction to compute Δh from Q. The solver is as willing to run it the other way, and in the inference problems below it is run both ways at once, since neither the flows nor the heads are known in advance.

20.2.1 Why a closed form exists

Two pipes in parallel span the same pair of nodes, so they lose the same head. Writing (20.1) for each and dividing eliminates that head:

$$K_1 Q_1^{1.852} = K_2 Q_2^{1.852}, \tag{20.2}$$

$$\frac{Q_1}{Q_2} = \left(\frac{K_2}{K_1}\right)^{1/1.852}. \tag{20.3}$$

The split ratio is fixed by the pipes alone, independent of how much water is flowing. Conservation at the two junctions then fixes the total, $Q_1 + Q_2 = d_1 + d_2 + \ell$, and with the ratio in hand every flow follows; the heads follow by substituting back into (20.1). For this network (20.3) predicts $Q_1/Q_2 = 2.852410$.

That is the whole reason this network was chosen. A five-unknown nonlinear system that a solver must iterate on has, here, an answer obtainable with a calculator, and any disagreement between the two is a defect in the solver rather than a feature of the physics.

20.2.2 One regularization, and what it costs

Law (20.1) has one numerical defect. Its derivative with respect to flow, which is what a Newton step needs, behaves as $Q^{0.852}$, which vanishes at $Q = 0$. A solver started from rest, or a pipe that genuinely carries nothing, meets an infinite slope in the inverse direction. The standard repair replaces the law by a smooth version that agrees with it away from zero:

$$\Delta h_e \;=\; K_e\, Q_e \left(Q_e^2 + q_0^2\right)^{0.426}, \tag{20.4}$$

with q_0 a small reference flow. Expanding for $Q \gg q_0$ shows the price: the relative departure from (20.1) is $0.426\,(q_0/Q)^2$, so the regularization is a second-order effect that any reported digit can be made insensitive to by choosing q_0 small enough. This chapter reports the state at two settings — a deliberately tiny $q_0 = 10^{-8}$ m^3/s and the default a practitioner would leave in place — so the reader can see the size of the effect rather than take it on trust.

Equation (20.4) also makes the law odd in Q, so flow reversal is handled without a case split. Chapter 7 returns to this: a smooth surrogate for a law with a kink is the cheapest of the four ways out of the Gaussian world, and the only one that costs nothing at solve time.

20.3 The model as a factor graph

The unknowns are the three pipe flows m_1, m_2, m_3 in kg/s and the two junction heads h_1, h_2 in meters. The reservoir head and both demands are known. Five relations tie the five unknowns: conservation at each of the two junctions, and the closure (20.4) on each of the three pipes,

$$r_1 = m_1 + m_2 - m_3 - d_1, \tag{20.5}$$
$$r_2 = m_3 - d_2 - \ell, \tag{20.6}$$
$$r_3 = h_R - h_1 - K_1 Q_1 (Q_1^2 + q_0^2)^{0.426}, \tag{20.7}$$
$$r_4 = h_R - h_1 - K_2 Q_2 (Q_2^2 + q_0^2)^{0.426}, \tag{20.8}$$
$$r_5 = h_1 - h_2 - K_3 Q_3 (Q_3^2 + q_0^2)^{0.426}, \tag{20.9}$$

where $Q_e = m_e/\rho$ and ℓ is the leak outflow, absent from the leak-free model. Each gauge adds a reading row, and each leak hypothesis adds one unknown and one prior row:

$$r = y_i - h_i \;\; (\sigma = 10 \text{ cm}), \qquad r = \ell \;\; (\tau = 20 \text{ kg/s}). \tag{20.10}$$

Figure 20.2 draws the graph. Two features of it carry the chapter.

First, the loop is visible: m_1 and m_2 both reach h_1, once through conservation and once through their own closures, so the graph has a cycle and the elimination of Chapter 8 must fill.

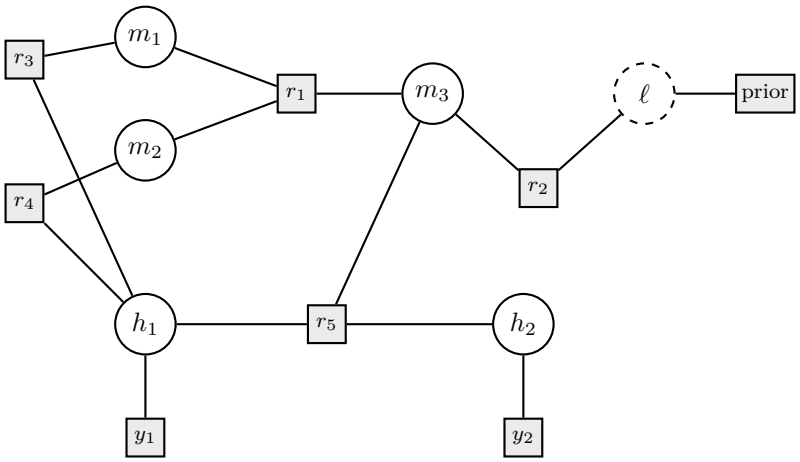


Figure 20.2: The district as a factor graph. Solid circles are unknowns, the dashed circle is present only under a leak hypothesis, and squares are rows: r_1, r_2 conservation at the two junctions, r_3, r_4, r_5 the three pipe closures, y_1, y_2 the gauges. Five unknowns and five physics rows make a square system when the leak is absent and the gauges are ignored; the two reading rows and, under a leak hypothesis, the extra unknown ℓ with its prior row, turn it into the estimation problem of Chapter 3. Only the dashed node and the two rows touching it change between the three hypotheses compared in Section 20.5.3.

On a network this size that is free. Chapter 5 explains why it is not free on a real one, and Chapter 21 measures the cost on a network of ninety-two junctions.

Second, the three hypotheses the chapter compares differ by almost nothing. A leak at J2 attaches ℓ to row r_2; a leak at J1 attaches it to r_1 instead; no leak deletes the node and its prior. Everything else — every closure, every conservation row, both gauges — is shared. This is the general shape of a diagnosis problem as Chapter 12 sets it up, and the reason the comparison is cheap: the models are not three models but one model with one row rewired.

> **Principle 20.1.** *A fault hypothesis is a graph edit, not a separate model. Detecting, sizing and locating a fault are the same inference run on graphs that differ by one node and one row.*

20.4 Method

Three questions, three operations on the graph of Figure 20.2, each justified in an earlier chapter.

The state. With the demands known, no leak and the gauges ignored, rows (20.5)–(20.9) are five equations in five unknowns. Newton's method from a flat start — heads at the reservoir head, flows at 1 kg/s — solves them. This is the deterministic solve of Chapter 1, and it is checked twice: against the closed form of Section 20.2.1, whose reference is machine precision, and against EPANET 2.2, the reference solver of the water industry, given the same network as an input file and run to an accuracy of 10^{-8}. EPANET's own output is written to about 10^{-4} m, which is the precision at which that second comparison can say anything.

The size. Admit the two gauge readings and one leak unknown at a named junction. Every row now carries a weight, the inverse variance of how far it may be off: the gauges at $\sigma = 10$ cm, the leak prior at $\tau = 20$ kg/s — wide enough to say nothing about the size — and the physics rows at 1 g/s on conservation and 1 mm on the closures, so that the physics is trusted some three orders of magnitude more than any reading. That choice is the softening of Chapter 4: the physics is not imposed exactly, but so tightly that the posterior lives in a thin shell around the manifold. The most probable state and the posterior width of ℓ follow from the Gauss–Newton solve and the Laplace approximation of Chapter 6, both read off the same sparse factorization.

Whether the model fits at all is a separate question from what it estimates, and is answered before it. The reading rows' weighted squared residuals at the optimum are compared against a chi-square distribution on as many degrees of freedom as there are readings minus fitted parameters, and the model is rejected at the one-percent level. Chapter 12 calls this the adequacy test, and insists it precede any estimate the model produces.

The location. Three graphs compete for the same two readings: no leak, a leak at J1, a leak at J2. Each is solved and each is scored by its evidence — the probability it assigns to the readings after integrating over its own unknowns, which in the Laplace approximation is

$$\log Z \;=\; -C(z^*) \;+\; \tfrac{d}{2}\log 2\pi \;-\; \tfrac{1}{2}\log\det \mathbf{\Lambda} \;+\; \sum_k \tfrac{1}{2}\log\frac{w_k}{2\pi}, \tag{20.11}$$

with C the weighted cost at the optimum, d the number of unknowns and $\mathbf{\Lambda}$ the information matrix. The final sum normalizes each row by its own weight, which is what lets graphs with different numbers of unknowns be compared as densities rather than as fits; without it a model that adds an unknown always wins. The third term is the Occam factor of Chapter 9: a model whose extra freedom is not needed pays for it here. Posterior probabilities are the evidences normalized under equal prior odds.

The threshold. The three-way comparison is then repeated with the true leak set to 0.25, 0.5, 1, 2 and 5 kg/s in turn, ten fresh noise draws each, with the gauges and their noise unchanged. This locates the leak size at which two gauges stop being able to answer.

20.5 Results

20.5.1 The state, four ways

Table 20.1 lists the state under each of the four routes. The first two columns agree to fifteen digits. The third differs in the sixth decimal of the head, and the size of that difference was predicted before it was measured: Section 20.2.2 said the regularization costs $0.426\,(q_0/Q)^2$ in relative terms, and at the default q_0 on a flow of 74 L/s that is the 5×10^{-6} m observed. The fourth agrees with the first to 0.4 mm, which is as close as EPANET's printed output can come.

The split ratio is the cleanest check in the table. Equation (20.3) predicts $Q_1/Q_2 = 2.852410$ from the two pipes' coefficients alone, with no reference to demands, heads or the third pipe. The solved flows give 2.852410.

> **Principle 20.2.** *Two independent references agree with the equations to the precision each can offer: the closed form to machine precision, the industry solver to its own output precision. Neither check would have caught a sign error that the other missed, and running only one of them would have been weaker than the reader needs.*

Table 20.1: The leak-free state, computed four ways. The Newton solve at $q_0 = 10^{-8}$ m^3/s reproduces the closed form to 4.0×10^{-15} relative, which is machine precision. The default regularization moves the heads by 4.9×10^{-6} m, the second-order effect predicted in Section 20.2.2. EPANET 2.2 differs from the closed form by 3.8×10^{-4} m in head and 5.4×10^{-9} m^3/s in flow, which is its own output precision.

quantity	closed form	Newton, $q_0 = 10^{-8}$	Newton, default q_0	EPANET 2.2
Q_1 [L/s]	74.0422	74.0422	74.0422	74.0422
Q_2 [L/s]	25.9578	25.9578	25.9578	25.9578
Q_3 [L/s]	40.0000	40.0000	40.0000	40.0000
h_1 [m]	93.97740	93.97740	93.97740	93.97760
h_2 [m]	85.70010	85.70010	85.70010	85.70040

A leak of 10 kg/s at J2 — ten percent of the water the district delivers — lowers h_2 by 5.40 m and h_1 by 1.16 m. The second of those is the interesting number. J1 has no leak, and a water balance drawn around J1 alone would show nothing wrong there, yet its gauge moves by nearly twelve times its own noise. The series element propagates the fault: more water passes through P3 to feed the leak, so P3 loses more head, and so does everything upstream of it. A method that looked for the leak by asking which gauge moved would be looking at two suspects.

20.5.2 How big, and whether the interval is honest

With the leak placed at J2 and its size left free, the estimate over twenty noise draws is 10.0441 kg/s against a true 10 kg/s, a bias of +0.044 kg/s, or four parts in a thousand. The mean posterior standard deviation is 0.167 kg/s.

The estimate is easy; the interval is the claim worth testing. A posterior standard deviation is a promise about how far the estimate will move if the noise is redrawn, and it can be checked directly by redrawing the noise. The spread of the estimate across the twenty draws is 0.205 kg/s against a mean posterior standard deviation of 0.167 kg/s — a ratio of 1.22. The posterior is too narrow by about a fifth. Consistently with that, 18 of the twenty 95 % intervals cover the truth where about 19 are expected.

Figure 20.3 shows every interval. Two features of it are worth naming. The misses are near-misses, not blunders: the estimate is biased slightly high and the two failures both fall on the same side. And the intervals are nearly identical in width, which is the Laplace approximation behaving as Chapter 6 says it should — the posterior width comes from the curvature of the cost, which depends on the network and the gauges, and only weakly on what the gauges happened to read.

Section 20.6 returns to the 1.22.

The adequacy test separates cleanly. On the first draw the leak-free model has a reading misfit of $\chi^2 = 3060$ on two degrees of freedom and is rejected; the model with a leak at J2 has $\chi^2 = 0.27$ on two degrees of freedom, $p = 0.87$, and is not. Across all twenty draws the leak-free model is rejected twenty times and the true-node model is adequate twenty times. This is worth separating from the estimate: before any number is quoted for the leak's size, the model that would quote it has been shown to be capable of fitting the data at all.

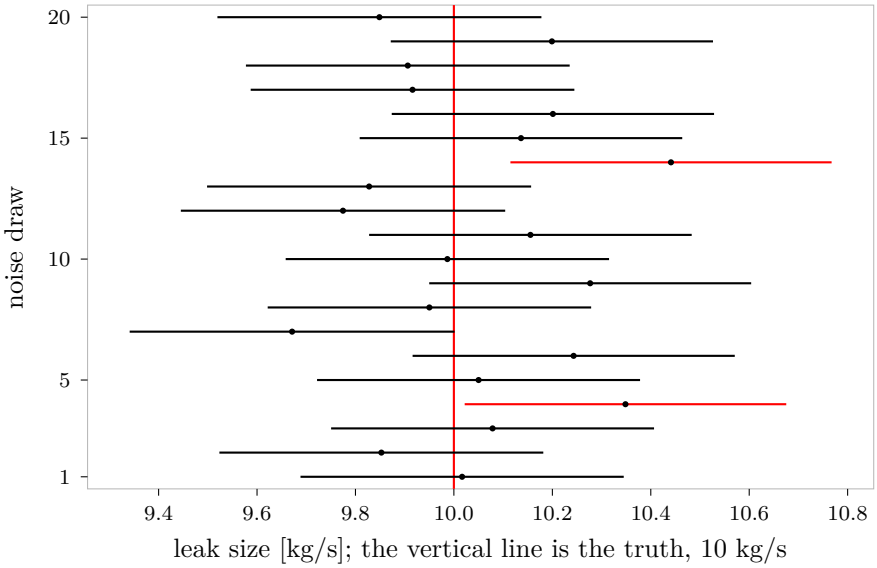


Figure 20.3: The twenty 95 % intervals on the leak's size, one per noise draw, against the truth. Two intervals (draws 4 and 14, drawn in red) miss; about one is expected. The intervals are of nearly equal width because the posterior width of a leak at a fixed junction depends on the gauges and the network, not on the readings.

20.5.3 Which junction

The evidence (20.11) puts J2 first in all twenty draws, with a posterior mass of 1.000 to four figures in every one of them. The margins are not close. Against no leak the smallest margin over the twenty draws is 1413 nats; against the neighboring junction it is 420 nats. A margin of five nats is already odds of about 150 to 1.

The first draw shows where the separation comes from. Its three log-evidences are -1526.85 for no leak, -455.43 for a leak at J1 and -1.69 for a leak at J2. The leak-free model is not merely worse; it cannot fit at all, and its score is dominated by a cost term that the other two do not pay. The leak at J1 can fit one gauge but must then miss the other, because a leak at J1 draws extra flow through P1 and P2 but not through P3, and so moves h_1 and h_2 in a ratio the readings do not show. The leaking junction is identified not by which gauge fell but by the ratio in which the two fell together.

> **Principle 20.3.** *A single sensor tells where a fault is not. A pair of sensors on either side of an element tells where it is, through the ratio of their responses, which the network fixes and the fault's location changes.*

20.5.4 Where two gauges stop seeing

Table 20.2 and Figure 20.4 give the sweep. The transition is narrow. At 0.25 kg/s the leak's signature at J2 is 0.124 m, about 1.2 gauge standard deviations, and the answer is wrong: the true junction ranks first in one draw of ten and the no-leak hypothesis carries 0.73 of the mass. At 0.5 kg/s the signature is 2.5 standard deviations and the answer is still wrong more often than right — the true junction leads in two draws of ten. At 1 kg/s the signature is 5.0 standard

Table 20.2: The detection sweep: the same three-way comparison at five true leak sizes, ten noise draws each. Rejection of the leak-free model and correct ranking of the true junction both turn over between 0.5 and 1 kg/s. The size estimate does not: it stays close to the truth at every size, including the sizes at which the location is unknown.

leak [kg/s]	signature at J2 [m]	leak-free rejected	true node ranked first	P(J2) mean	P(none) mean	estimate [kg/s]
0.25	−0.124	2/10	1/10	0.111	0.729	0.306
0.5	−0.249	3/10	2/10	0.274	0.450	0.556
1	−0.499	10/10	10/10	0.829	0.036	1.055
2	−1.008	10/10	10/10	1.000	0.000	2.054
5	−2.587	10/10	10/10	1.000	0.000	5.052

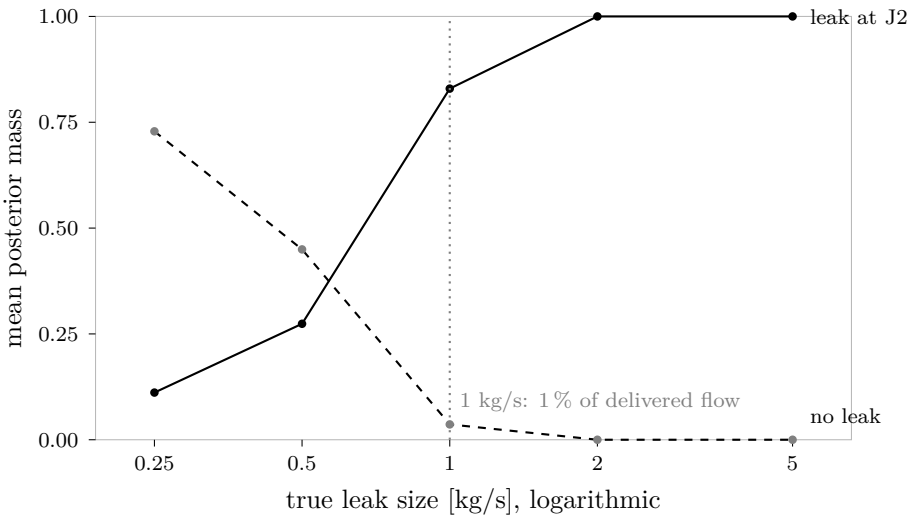


Figure 20.4: Mean posterior mass on the true junction (solid) and on no leak (dashed) against the true leak size. The crossing is sharp: at 0.5 kg/s the no-leak hypothesis still holds more mass than the truth; at 1 kg/s, one percent of the district's delivered flow, the truth holds 0.83 and is ranked first in every draw.

deviations, and the answer is right in every draw, with 0.83 of the posterior mass and the leak-free model rejected ten times out of ten.

One kilogram per second is one percent of what this district delivers. That is the honest capability statement for two gauges of this quality on this network: a leak of one percent of throughput is found reliably, a leak of half a percent is not found at all, and nothing about the method changes that — the information is not in the readings.

20.5.5 Cost

Every number above costs 218 engine solves: 8 Newton solves for the deterministic states of Table 20.1 and the true states behind each sweep point, and 210 Gauss–Newton solves with their Laplace posteriors — twenty draws and fifty sweep draws, each against three hypotheses, which is $60 + 150$. The whole study runs in 9.1 seconds.

The count is quoted in solves rather than seconds because it is the quantity that does not depend on the machine, and because it is the quantity that scales. Chapter 22 asks the same location question of 268 junctions rather than two, where evaluating every hypothesis exactly is no longer free, and the screening argument that makes it affordable is the subject of Chapter 9.

20.6 What surprised

The posterior is too narrow, and by a knowable amount. The ratio of the observed spread to the promised standard deviation is 1.22, not 1.00. This is the Laplace approximation being asked to describe a posterior that is not quite Gaussian: the leak enters the conservation row linearly but changes the flow in P3, which enters the closure rows through a power law, so the cost is not exactly quadratic in ℓ and its curvature at the optimum slightly overstates how sharply the cost rises away from it. The effect is small here and would be smaller still for a smaller leak, since the nonlinearity is in the flow change. It is reported rather than absorbed because a twenty-percent optimism in an interval is exactly the size of error that goes unnoticed when nobody redraws the noise. Chapter 7 gives the general diagnosis, and Chapter 23 finds the same failure at a far larger scale, where it matters.

Sizing survives below the detection threshold. The sweep was run to find where the method stops working, on the assumption that it would stop working all at once. It does not. At 0.25 kg/s the location is unknown — one correct ranking in ten draws, most of the mass on no leak — while the size estimate, *conditioned on the leak being at J2*, is 0.306 kg/s against a truth of 0.25, and nine of ten intervals still cover. The two questions degrade at different rates because they use different parts of the information. Sizing uses the magnitude of the head drop, which is proportional to the leak and stays measurable; locating uses the ratio of the two gauges' responses, which is a difference of two small numbers and is destroyed by noise much sooner.

The practical consequence is a trap. A tool that reports a size without reporting the location's confidence will report 0.306 kg/s at a junction chosen essentially at random, and the number will look reasonable. This is why Section 20.5.2 runs the adequacy test before quoting the estimate, and why the location is reported as a distribution over junctions rather than as the best one.

The leak-free model's rejection is not the same event as the leak being located. At 0.5 kg/s the leak-free model is rejected in three draws of ten while the true junction leads in two. Rejection says the model is inadequate; it does not say which replacement is right. At small leak sizes there is a band in which an operator correctly learns that something is wrong and cannot yet be told where. On this network that band is narrow. On a real one, with more junctions competing and fewer gauges, it is the normal operating condition, and Chapter 22 works inside it.

20.7 Reproduction

The study runs in about nine seconds on a workstation:

```
python docs/terra_graph_engine/case_studies/water_networks/\
       parallel_pipes_and_a_leak/run.py
```

It writes `results.json` and `timings.json` beside itself. The chapter's numbers and figures are then reproduced from that file alone:

```
cd docs/books/hydraulic && python scripts/w0_smallest_loop.py
```

which prints every value quoted above and rewrites the four fragments in `figures/`. A representative line of its output:

```
  spread of the estimate across seeds   0.204767 kg/s
  mean posterior standard deviation     0.167451 kg/s
  spread / posterior sd                 1.2228    <- 1.00 if calibrated
```

The EPANET column of Table 20.1 needs EPANET 2.2 through WNTR 1.5.0; the remaining columns need neither.

Notes

The Hazen–Williams law and its coefficient (20.1) are standard and appear in every water-distribution text. The regularization (20.4) that keeps its slope finite at zero flow is the device EPANET itself uses for low flows; the form here, and the $0.426\,(q_0/Q)^2$ error bound, are stated so the reader can choose q_0 rather than inherit it. The closed form for parallel pipes is elementary.

Writing a water network's conservation and head-loss relations as a factor graph and estimating its state from sparse readings is the construction of Han et al. [33] for an earthquake-damaged network and of Irofti et al. [35] for leak localization over a time window; both rest on the sparse least-squares machinery described by Dellaert and Kaess [17]. Comparing fault hypotheses by their Bayesian evidence rather than by a score is older than all three and is treated in general in Chapter 9. The second of the two checks in Section 20.5.1 is EPANET 2.2, whose hydraulic formulation is the one this chapter's rows reproduce.

What is this chapter's own is the pairing of the two independent checks — a closed form and an industry solver on the same five-unknown system — and the observation of Section 20.6 that sizing and locating degrade at different leak sizes, which does not appear to be stated in the leak-detection literature even though it follows directly from which part of the information each question consumes.

Chapter 21

Half the pipes have meters

Numbers in this chapter are produced by `scripts/w1_net3.py` from the study's committed results file. Nothing is retyped.

21.1 The question

Chapter 20 solved a network with five unknowns, and every claim it made was checked against a closed form. This chapter's network has two hundred and eleven, and there is no closed form. A utility in this position has an estimate of its own state and no way to confirm it: the quantity it most wants to know — the head at an unmetered junction — is unmetered precisely because nobody measures it.

So the question is not "is the answer right." It is: *what can be checked instead?* This chapter checks four things, and they are chosen so that each one could fail independently of the others.

1. **Is the error bar honest?** The estimate comes with a posterior standard deviation at every junction. That is a falsifiable claim about how far the truth will be, and it can be tested by generating truths.
2. **Is this the standard estimator?** Water-network state estimation has an established form — weighted least squares. If the construction of this book gives a different answer, the difference has to be explained, and if it gives the same one, the probabilistic machinery has to be earning something else.
3. **Does the decomposition work?** The network has articulation points, and the literature cuts networks there. Does cutting there actually separate the computation?
4. **Does it survive a hard task?** Twelve pipes are broken at once. Finding them is a question the calibration checks cannot answer.

Three of the four come back clean. The third does not do what it is expected to, and the fourth does worse than a published number — for a reason that turns out to be a property of the sensor layout rather than a defect in the method.

> **Principle 21.1.** *On a system too large to verify, verify properties of the answer rather than the answer: that its stated uncertainty is calibrated, that it agrees with the established estimator where one exists, that its decomposition is exact, and that it degrades in ways the model itself predicts.*

21.2 The system

Figure 21.1 draws it. EPA Net3 is EPANET's public example network: 92 junctions, 58 of them delivering water to consumers, fed by two reservoirs and three tanks held at fixed heads between 42.7 and 67.1 m. At the snapshot used here one pipe and one pump are closed, leaving 117 open links on 97 nodes. The nominal demand is 680 kg/s.

The instrumentation is deliberately thin, and matches what a real utility has:

- flow meters on half the open pipes — 58 of 116 — each with a standard deviation of 1 kg/s, and *a different random half in every draw*, which turns out to matter more than anything else in the chapter;
- exactly one head gauge, at junction 61, with a standard deviation of 1 cm;
- every junction demand known only to 30 percent of its nominal value, with a floor of 0.5 kg/s.

The last of these is the important one and it is easy to skim past. The demands are not inputs here; they are *unknowns with priors*. A utility does not know how much water each street took this hour. It knows roughly, and roughly is a distribution. Section 21.5.4 shows what that costs.

21.3 The model as a factor graph

There is nothing new in it. The unknowns are a head at each of the 92 junctions, a signed mass flow on each link, and a demand at each demand-carrying junction — 211 in all. The rows are the rows of Chapter 20: one conservation relation per junction, one Hazen–Williams closure per pipe, the running pump's curve, one reading row per meter, and one prior row per uncertain demand. The fixed heads are substituted constants; a closed link's flow is held at zero by its own row.

That the model is unchanged is worth stating rather than passing over. The step from two junctions to ninety-two adds no construction. It adds only the thing this chapter is about: the loss of the ability to check.

21.4 Method

Calibration. Twenty draws. In each, a fresh random half of the pipes gets meters, the demands are drawn from their priors, and an independent simulator solves the network exactly with those demands to give the truth. The readings are that truth plus their noise. The estimate is then compared against the truth at all 92 junctions, and the count of junctions whose truth falls within two and three posterior standard deviations is what a calibrated posterior makes a prediction about: 95.45 and 99.73 percent. The reading residuals are also tested against the stated noise by the chi-square misfit of Chapter 20, at the one-percent level.

Weighted least squares. With the demands of one draw treated as known, the network's state is over-determined by the meters and nothing is left for a prior to do. The most probable state is then solved twice: once by the Gauss–Newton iteration of Chapter 6, and once by an independent optimizer running Levenberg–Marquardt on the same weighted residuals. If the two disagree, one of them is wrong.

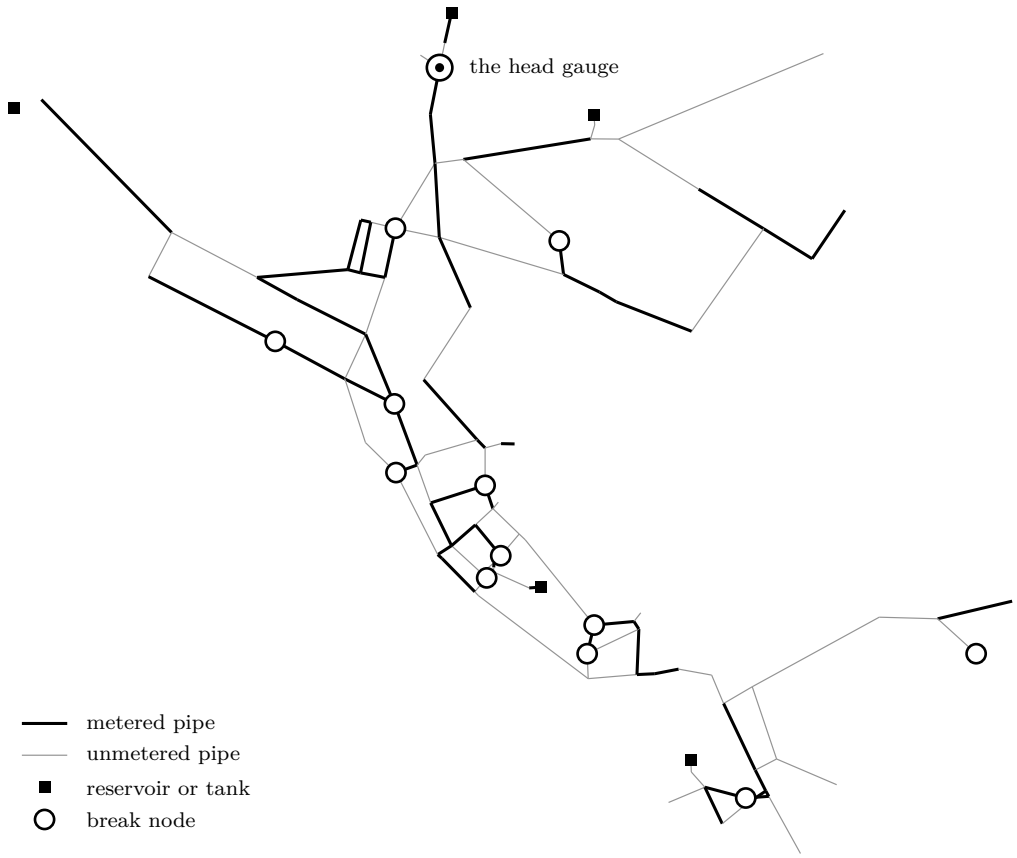


Figure 21.1: EPA Net3: 92 junctions, of which 58 carry a demand, fed by two reservoirs and three tanks at fixed head, on 117 open links. Half the pipes carry a flow meter — a fresh random half in each of the twenty draws, the set shown being the first — and a single head gauge sits at one junction. The twelve circled nodes are where pipes are broken in Section 21.5.4. Nothing in the figure marks which junctions the estimate is good at, which is the point of the chapter.

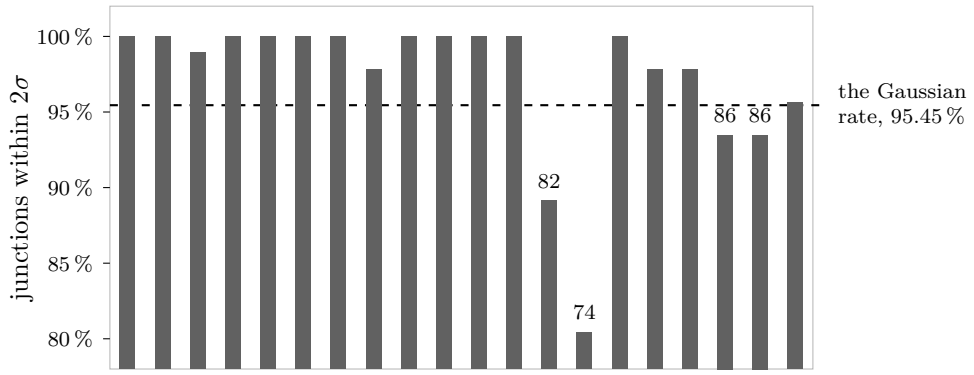


Figure 21.2: The same coverage, per draw. Eleven of the twenty draws cover every one of the 92 junctions within 2σ; the numbers above the short bars count the junctions covered in the draws that do not. The pooled figure of 97.23 percent is an average over a population with two very different kinds of member.

The decomposition. The open-link graph's articulation points — the nodes whose removal disconnects it — are computed, along with its biconnected components and bridges. Three candidate interfaces are then formed, and for each one the information matrix is permuted into interface and interior and the interior's block structure counted. The marginal covariance of the interface is computed twice, once as the inverse Schur complement and once as the corresponding block of the full inverse, and the two are differenced. An elimination-cost predictor scores each interface against solving the whole thing at once.

Breaks. Twelve of the 116 open pipes are broken at random, each adding a 5 kg/s outflow at a junction end: 60 kg/s in all, 8.8 percent of the network's demand. The demands stay unknown and the meters stay at half the pipes. Two routes then rank the junctions:

- the *continuous* route gives every junction a latent outflow at once, solves the enlarged model *once*, and ranks junctions by the z-score of their fitted outflow;
- the *hybrid* route builds one model per junction, each with a single outflow at that junction, solves all 93 of them — the 92 hypotheses and the leak-free model — and ranks by evidence gain over the leak-free model, exactly as Chapter 20 ranked its three hypotheses.

One costs one solve and the other ninety-three; the comparison is what that buys.

21.5 Results

21.5.1 Is the error bar honest?

Pooled over the 1840 (draw, junction) pairs, 1789 truths fall within two posterior standard deviations and 1839 within three — 97.23 and 99.95 percent, against the 95.45 and 99.73 a calibrated Gaussian predicts. The head root-mean-square error is 5.4 cm. The misfit test accepts all twenty draws.

Read as a single line that is a good result, slightly conservative. It is also, on its own, misleading.

Figure 21.2 splits it by draw. Eleven of the twenty draws have *no* junction outside 2σ at all. One draw has eighteen. The failures per draw run

$$0,\ 0,\ 1,\ 0,\ 0,\ 0,\ 0,\ 2,\ 0,\ 0,\ 0,\ 0,\ 10,\ 18,\ 0,\ 2,\ 2,\ 6,\ 6,\ 4.$$

If the junctions failed independently at the Gaussian rate, each draw would have about 4.2 failures with a variance of 4.0. The observed mean is 2.55 — lower, which is the conservatism the pooled number reports — but the observed variance is 20.8, a dispersion ratio of 5.2. Failures are not independent. They arrive in clusters.

The cause is the one thing that changes between draws: the meter set. A draw that happens to leave a neighbourhood unmetered has no information about the heads in it, and every junction there is estimated from demand priors alone — badly, and with an error bar that does not widen enough to cover the damage. The failures are spatially correlated because the cause is.

Principle 21.2. *Coverage pooled across draws is the wrong statistic when the failures are correlated within a draw. A posterior that is conservative on average can still be badly wrong everywhere at once, in the cases that matter.*

This is not an abstraction for a utility. The pooled number says "the error bars are fine." The per-draw picture says "with this particular meter layout they are fine, and with that one an entire district is wrong at the same time." The second is the sentence an operator needs.

21.5.2 It is weighted least squares

With the demands known, the most probable state and the independently computed weighted-least-squares state agree to below 10^{-9} in both head and flow. (The recorded figures are 7.1×10^{-15} m and 7.5×10^{-13} kg/s; they move with the machine's linear-algebra kernel, so the claim is the bound, not the digits.) Against the simulator's own truth for that draw, the root-mean-square head error is 9.5×10^{-5} m, which is the residue of meter noise on a fully determined state.

This is the answer to the second question, and it is deliberately a *negative* result: the construction of this book produces exactly the estimate the field already produces. Physics rows weighted very tightly, reading rows weighted at their sensor noise, and no priors — that is weighted least squares, and Chapter 6 derives the identity rather than asserting it.

What the probabilistic framing adds is everything around the point estimate: the covariance of Section 21.5.1, the demand priors that make the under-determined problem well-posed at all, and the model comparison of Section 21.5.4. None of those is available from the minimization alone. A reader who wants the standard answer gets the standard answer; a reader who wants an error bar has to have come this way.

21.5.3 What the cut separates

Net3's open-link graph has 31 articulation points, all of them junctions. Removing them leaves 35 biconnected components: one large one of 56 nodes, then components of 6 and 4, and 32 that are single links — bridges. This is real structure, and it is where the literature cuts.

So cut there. Take the 31 articulation heads as the interface, permute the information matrix, and count the blocks the interior falls into.

It falls into **one**.

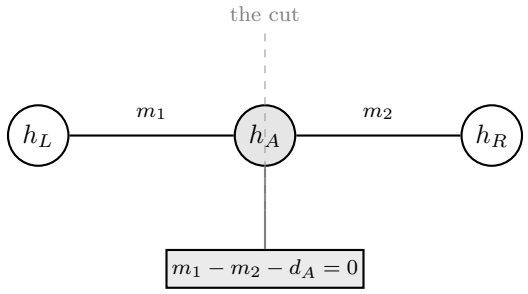


Figure 21.3: Why the graph cut is not the matrix cut. Junction A is an articulation point: deleting it disconnects the left side from the right. But deleting the *variable* h_A does not delete the conservation *row* at A, and that row names the flows on both incident links. Left and right stay coupled through m_1 and m_2. The interface has to take the flows as well.

Table 21.1: Three interfaces on the same network. Only the second and third separate anything. In every case the interface's marginal covariance computed as an inverse Schur complement equals the corresponding block of the full inverse to round-off, so the choice of interface does not affect the answer — only whether the interior factorizes.

interface	size	interior blocks	Schur vs full inverse [m^2]	cost ratio
heads only	31	1	3.1e-18	0.237
heads + the 32 bridge flows	63	18	3.7e-14	0.261
heads + all 64 incident flows	95	19	4.9e-13	0.326

Figure 21.3 is the reason, and it is one line once seen. An articulation point separates the graph by *node* removal. The information matrix is a matrix over *variables*, and a junction contributes two kinds — its head, and the flows on the links meeting there. Removing the head from the interior leaves the conservation row behind, and that row still names the flows on both sides of the cut. The two sides remain tied together by a variable the cut never touched.

Table 21.1 gives all three choices. Adding the 32 bridge flows to the interface — taking it from 31 variables to 63 — takes the interior from one block to eighteen. Adding all 64 flows incident to articulation junctions gives nineteen, at an interface of 95.

Two things in that table are worth separating.

The first is that the Schur complement is exact in all three cases. The interface's marginal covariance, computed by eliminating the interior, equals the corresponding block of the full inverse to between 10^{-18} and 10^{-13} m^2. This is Chapter 8's claim about the Schur complement being a message, verified on a real network rather than a chain of three nodes. *Whichever* interface is chosen, the answer is the same; the interface decides only the cost.

The second is that the useful cut is not the cheapest one. By the elimination predictor, the head-only interface scores 0.237 of the monolithic cost and the bridge-flow interface 0.261 — ten percent worse. But the head-only interface has one interior block, which means it is not a decomposition at all: nothing has been separated, and there is nothing to solve independently, in

Table 21.2: The twelve break nodes. Each carries a 5 kg/s outflow. “Prior sd” is the standard deviation of that junction’s own demand prior, which is 30 percent of its nominal demand; “metered” counts its incident links that carry a flow meter. The last two columns are the node’s rank among the 92 junctions under each route.

node	demand [kg/s]	prior sd [kg/s]	degree	metered links	rank continuous	rank hybrid
109	19.56	5.87	2	1	32	37
113	1.69	0.51	3	1	7	16
120	0.00	0.00	4	2	6	32
153	3.73	1.12	2	0	82	53
183	0.00	0.00	3	0	54	4
191	6.92	2.08	3	1	30	23
199	10.09	3.03	3	2	11	25
225	1.93	0.58	1	1	1	1
247	5.95	1.78	3	0	72	27
265	0.00	0.00	3	1	34	7
271	0.00	0.00	3	1	59	11
273	0.00	0.00	3	2	2	2

parallel, or to leave alone when a reading changes on the other side of the network. Ten percent is what the eighteen blocks cost.

Principle 21.3. *A node separator of the graph is a variable separator of the factor graph only when the factors at the separating nodes are split with it. Cutting a network at its articulation points is necessary and not sufficient; the flows crossing the cut are variables too.*

21.5.4 Where are the breaks?

Both routes put five of the twelve break nodes in their top twelve. The hybrid route finds two more in its top twenty-four; the continuous route finds none. That is a recall of 41.7 percent against the 80 percent a published study reports for its own broken-pipe task on this network. The honest summary is that this is worse.

The interesting part is *which* seven are missed, because Table 21.2 explains every one of them, and the explanation is not about ranking.

A leak at a junction has to compete with that junction’s own demand prior. Both are unmetered outflows at the same node; nothing in the physics distinguishes them. So a 5 kg/s leak is visible only if the demand prior is narrower than 5 kg/s — otherwise the model can absorb the entire leak as an unremarkable demand fluctuation and pay almost nothing in evidence for it.

Junction 109 is the clean case. Its nominal demand is 19.56 kg/s, so its prior standard deviation is 30 percent of that, 5.87 kg/s — wider than the leak sitting on it. Both routes rank it 32nd and 37th, deep in the noise. It is not that the method failed to find it; it is that a 5 kg/s outflow at a node uncertain to 5.9 kg/s *is* a plausible demand, and saying otherwise would be a claim the data does not support.

Of the eleven break nodes whose prior is narrower than the leak, eight are found by at least one route. Of the one whose prior is as wide as the leak, none is.

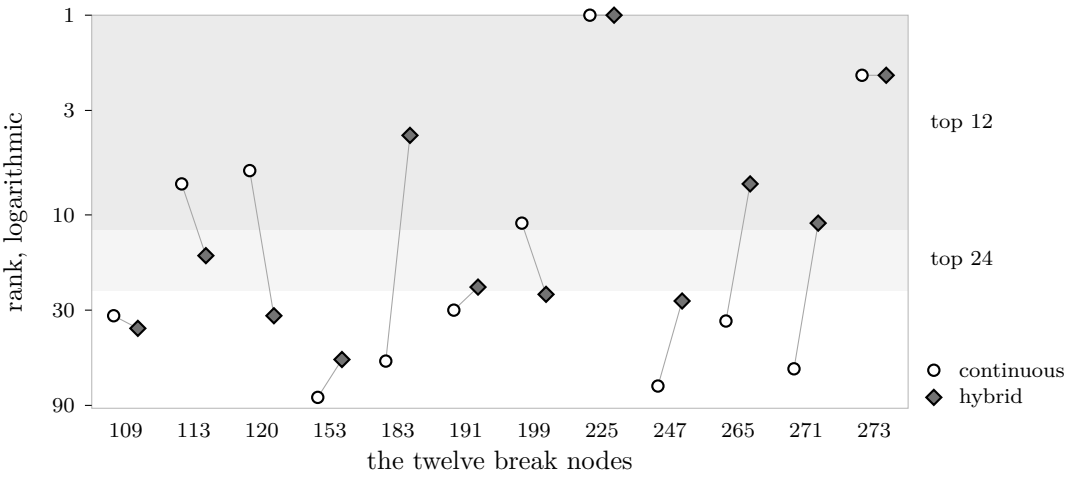


Figure 21.4: The same ranks, drawn. Each break node contributes a circle (the one-solve continuous route) joined to a diamond (the ninety-three-solve hybrid route); the shorter the line, the more the two routes agree about that node. The darker band is the top twelve — the number of breaks, and so what a utility sending out twelve crews would act on — and the lighter band the top twenty-four. The two routes disagree violently about individual nodes while scoring almost the same in aggregate.

The second mechanism is metering. Of the three break nodes with no metered link incident to them, one is found; of the nine with at least one, seven are. An outflow with no nearby meter changes heads that nobody reads.

Principle 21.4. *A fault is identifiable only against the prior it competes with. When a latent fault and an uncertain input enter the same conservation row, the fault is detectable to the extent that it is larger than the input's uncertainty — and no ranking statistic can recover what the priors have already absorbed.*

Two further details say that the method is failing gracefully rather than randomly. Every one of the hybrid route's seven false positives is within three hops of a true break, and the continuous route's within five: the evidence lands in the right neighbourhood even when it names the wrong node. And three of the hybrid route's candidates — junctions 163, 164 and 166 — carry the same evidence gain, 11.18 nats, to six digits. They lie on one unmetered path. They are not three hypotheses that the ranking got wrong; they are one hypothesis, and no amount of ranking can split a tie the sensors created. That is the same structural non-identifiability Chapter 22 meets on a larger network.

The presence of breaks is not in doubt. The leak-free model ranks fiftieth of the ninety-three hypotheses and the top hypothesis carries posterior probability 0.981. The network is certainly leaking; where is the open question.

21.5.5 Two hypotheses that converge slowly

Ninety-one of the hybrid route's hypotheses converge in a median of twelve Gauss–Newton iterations. Two — junctions 241 and 243 — take more than a hundred (the recorded counts are 295 and 267, and they move by a few between linear-algebra kernels).

Both lie on a dead-end pipe carrying essentially no flow. That is exactly where the regularized Hazen–Williams law of Chapter 20 has its largest curvature, and it is also where a hypothesized outflow is nearly collinear with the dead end's own demand. Gauss–Newton drops the second-order term of the Hessian, which is harmless when residuals are small or the model is nearly linear and is neither here; convergence degrades from quadratic to linear, the gradient shrinking a few percent per iteration. The solves reach the same tolerance as the rest. They simply cost twenty times more, in the one place where the answer is least identifiable anyway.

21.5.6 Cost

Every number above costs 119 engine solves: one Newton solve, and 118 Gauss–Newton solves with their Laplace posteriors — twenty for the calibration draws, ninety-three for the hybrid route, one for the continuous route, and four for the remaining comparisons. A further 22 runs of the independent simulator produce the truths and the reference states; those are not engine solves and are counted separately. The study takes 41 seconds, two thirds of it in the break scenario.

The ratio worth carrying away is the last one. The continuous route costs one solve and the hybrid ninety-three, and they score the same at $k = 12$. The hybrid's ninety-three buy two extra nodes at $k = 24$ and, more usefully, false positives that sit closer to real breaks.

21.6 What surprised

The graph cut is not the matrix cut. Articulation points are the standard place to decompose a water network, and the expectation going in was that cutting at the thirty-one of them would separate the problem into the thirty-five biconnected components the graph has. It separates it into one piece. The flows are variables, the conservation rows at the cut nodes survive the cut, and the decomposition only appears once thirty-two more variables join the interface. This is a small result with a general shape: a structural intuition drawn on the network diagram does not automatically transfer to the matrix, because the diagram has nodes where the matrix has node *and* edge unknowns.

The pooled coverage number is the wrong number. 97.23 percent within 2σ reads as a mild conservatism. The per-draw dispersion is 5.2 times binomial: eleven draws are perfect and one has eighteen junctions outside 2σ simultaneously. Averaging a calibration over draws that differ in *which sensors exist* hides the only failure mode an operator cares about, which is not "a few junctions are occasionally wrong" but "this month's meter outage makes a whole district wrong at once."

This is now the third chapter in a row in which the point estimate behaves and the *uncertainty* is where the interesting behavior is. Chapter 20 found a posterior twenty percent too narrow; A companion study on a heating loop found one up to five times too wide; this one finds a posterior that is correct on average and clustered in its failures. Three networks, three different miscalibrations, all of them invisible unless the noise is redrawn and the result is looked at per draw rather than pooled.

A worse recall number, and it is the right answer. Forty-two percent against a published eighty is the kind of result that invites a better ranker. Table 21.2 says not to build one. Seven

of the twelve misses are accounted for by two facts about the *problem* — a demand prior wider than the leak, or no meter near it — and neither is something a ranking statistic can undo. The leak at junction 109 is not hidden by a weak method; it is hidden by the utility not knowing its own demand at that node to better than 5.9 kg/s. The fix is a meter, not an algorithm, and the model is what says so.

21.7 Reproduction

The study takes about forty seconds:

```
python docs/terra_graph_engine/case_studies/water_networks/\
       state_estimation_net3/run.py
```

It writes `results.json` and `timings.json` beside itself. The chapter's numbers and figures are then reproduced from those files alone:

```
cd docs/books/hydraulic && python scripts/w1_net3.py
```

which prints every value quoted above — including the per-draw failure counts and the dispersion ratio of Section 21.5.1, which the script derives from the recorded per-draw counts — and rewrites the five fragments in `figures/`. A representative line of its output:

```
variance across draws                20.787   (binomial would be 3.996)
dispersion ratio                     5.20   <- 1.00 if independent
```

The truths, the reference states and the weighted-least-squares comparison need EPANET 2.2 through WNTR 1.5.0 and a general-purpose optimizer; the decomposition and the break ranking need neither.

Notes

The network is EPANET's distributed example, and the demand-uncertainty setting, the meter fraction and the broken-pipe task follow Han et al. [33], whose decomposition of a damaged water network at its articulation points is the direct precedent for Section 21.5.3 and whose reported figures are what Sections 21.5.1 and 21.5.4 are read against. Those comparisons are against their published metrics on a different sensor set, not replications of their experiment. Weighted least squares as the standard form of network state estimation is older and general; the identity with the most probable state under tight physics weights is derived in Chapter 6. Articulation points, biconnected components and bridges are standard graph theory; the Schur complement as an exact marginal is Chapter 8, and the sparse least-squares machinery underneath is Dellaert and Kaess [17].

What is this chapter's own is Section 21.5.3's measurement — that the articulation-point cut leaves the information matrix in one block until the bridge flows join the interface, with the exactness of the Schur marginal verified separately for each interface — and Section 21.5.1's observation that the pooled coverage of a randomized-sensor experiment conceals a fivefold over-dispersion, so that the calibration must be read per draw. The identifiability argument of Section 21.5.4, that a latent fault competes with the prior in the same conservation row, is implicit in any Bayesian treatment; what is new here is the measurement of it against a known ground truth on every break node at once.

Chapter 22

Screen, then confirm

Numbers in this chapter are produced by `scripts/w2_modena.py` from the study's committed results file. Nothing is retyped.

22.1 The question

Chapter 21 ranked ninety-two leak hypotheses by solving ninety-three models. That is affordable once. It is not a method: the next network has two hundred and sixty-eight junctions, each hypothesis is a joint model over a window of snapshots rather than a single one, and the whole thing has to run for every scenario a utility wants to test.

So this chapter asks a cost question and a resolution question, and they turn out to be different questions.

- **Can most hypotheses be eliminated for far less than the cost of evaluating one?** If a shortlist can be produced cheaply and the exact evidence reserved for what survives, the ninety-three-solve approach scales.
- **Once the shortlist is exact, how close is the answer?** Against four published methods on the same benchmark, on the same scenarios.

There is a third thread running underneath both, and it is the one Chapter 21 left open. A leak at a junction and an error in that junction's demand are the same thing to the physics: both are unmetered outflow in the same conservation row. The only thing that separates them is that a leak is *constant* and a demand is not. That fact is what a window of snapshots exists to exploit, and Sections 22.5.2 and 22.5.3 measure exactly what it is worth.

22.2 The two networks

The chapter works on two, because the method and its limits are best seen at different scales.

The T-Example has nine junctions, ten pipes and one reservoir, with pressure and demand read at junctions 2, 4, 5 and at the reservoir — four of ten possible places. Its day is sampled at 289 steps. A single junction leaks 0.5 L/s, about six parts in a thousand of the mean inflow. Nine leak junctions times nine noise draws give 81 cases, each solved over a window of 24 snapshots. Heads are read to 5 mm and meters to one percent.

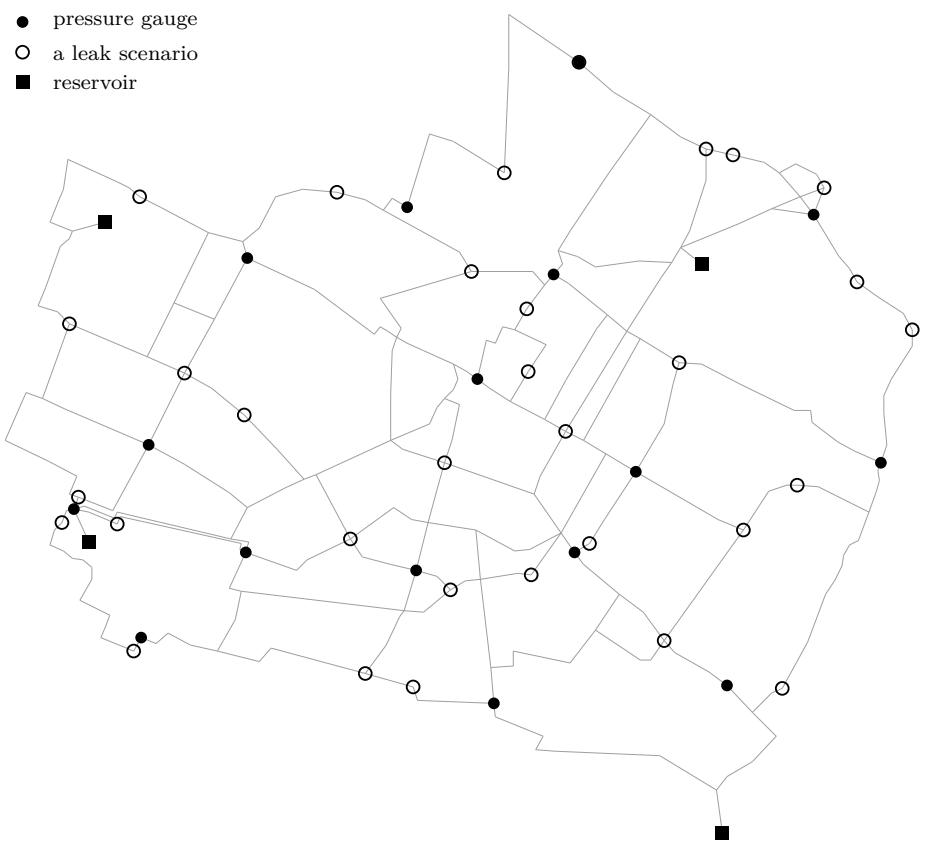


Figure 22.1: Modena: 268 junctions on 317 pipes, with twenty pressure gauges and the thirty-two junctions that leak, one per scenario. This is the density of instrumentation the chapter is working against — one gauge for roughly every thirteen junctions — and the reason a shortlist is necessary before any exact comparison.

The network has a feature worth naming before any result: junctions 7 and 8 lie on one unmetered path, as do 6 and 9. They are *structurally tied*. No reading distinguishes them, and Section 22.5.1 shows what that does to a probability.

Modena is the same file format at 268 junctions and 317 pipes, fed by four reservoirs, with 20 pressure gauges and 40 demand meters and 32 leak scenarios of 4.5 to 6.5 L/s. Two hundred and forty-eight of its junctions carry no pressure gauge at all. Its window is six snapshots and its heads are read to 1 cm.

22.3 The model: one unknown, many snapshots

For a window of T snapshots the model holds T copies of the network: heads at the junctions, flows on the pipes, and one demand variable per junction per snapshot, each with a Gaussian prior centered on that snapshot's nominal demand at half a percent of it. A hypothesis *leak at* j adds **one** variable — not T of them — to junction j's conservation row in every copy, with a zero-mean prior of 2 kg/s. Where j carries a demand meter, the meter reads demand plus leak, an exact auxiliary row. Figure 22.2 draws it.

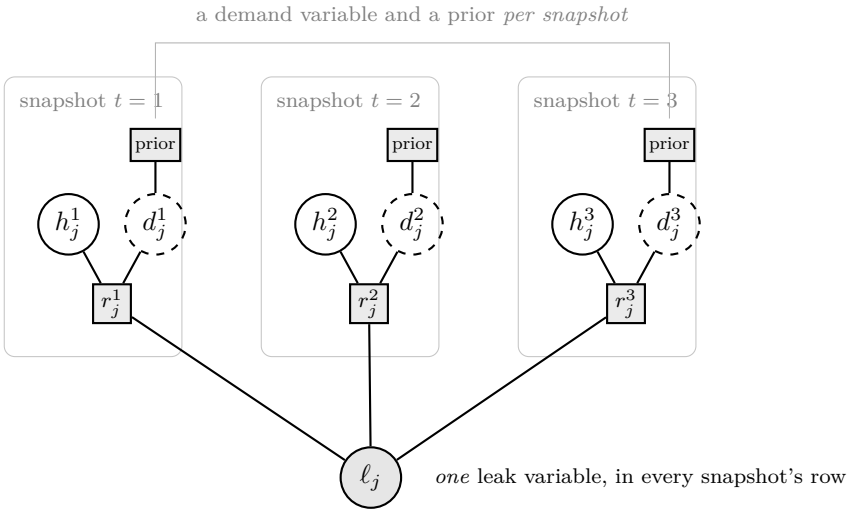


Figure 22.2: The window, drawn at one junction. Each snapshot carries its own head and its own demand, and the demand has its own prior in each — it is allowed to be different at nine in the morning and at three in the afternoon. The leak is *one* variable, entering the conservation row of every snapshot. That asymmetry is the entire source of the chapter's information: a genuine leak explains all T snapshots with one number, while a demand error would need T numbers, and pays for each of them.

> **Principle 22.1.** *Two explanations that are indistinguishable in one snapshot separate over many when one of them is constrained to be constant. The leak variable is shared across the window; the demand variables are not. Everything a window buys comes from that asymmetry, and nothing comes from having more data as such.*

22.4 Method

The exact comparison. Every hypothesis is solved to its most probable state and scored by evidence, exactly as in Chapter 20. Posterior probabilities are the evidences normalized under equal prior odds.

The screen. Solving 268 joint models per scenario is what the chapter is trying to avoid. The leak-free optimum already carries most of the answer, so a leak variable is added at junction j and *one* Gauss–Newton step of the augmented system is taken, eliminated onto the new variable against the already-computed factorization of the leak-free information matrix. That gives the leak the junction would take, the information it would carry and the evidence it would gain — a Rao score test:

$$\hat{\theta}_j = \frac{\sum_r w_r F_r}{S_j}, \qquad S_j = \sum_r w_r + \tau^{-2} - g_j^\top \mathbf{\Lambda}_0^{-1} g_j, \tag{22.1}$$

$$\Delta \log Z_j \approx \tfrac{1}{2}\,\hat{\theta}_j^2 S_j - \tfrac{1}{2}\log\!\left(S_j \tau^2\right), \tag{22.2}$$

where r runs over junction j's conservation factors in the T copies, F_r and w_r are their residuals and weights at the leak-free optimum, and g_j is the corresponding gradient. This costs one sparse

solve per junction against a factorization that has already been paid for, rather than one full nonlinear solve per junction.

Equation (22.2) is exact for a linear model, and on unmetered junctions it lands within 0.05 nats of the exact evidence. On *metered* junctions it over-scores, because the exact hypothesis also changes what the meter reads and the one-step approximation does not see that. So the screen does not decide. It shortlists — the twenty best unmetered and the five best metered candidates, twenty-five of 268 — and the exact evidence ranks what survives.

Calibration. A model whose pipe roughness is wrong expects the wrong head drop for a given flow, and a leak search will book the difference to a leak. Section 22.5.4 measures that. On Modena the roughnesses are therefore calibrated first, on six leak-free snapshots, each C_e a variable shared across the window with a prior of two percent about its nominal value, then held at its most probable value inside every leak hypothesis.

22.5 Results

22.5.1 Which junction leaks

Over the 81 T-Example cases the true junction is ranked first in 63, and first *or in an exact tie* in all 81. It is never below second. The median posterior on the true node is 0.959, the leak-free model is excluded by five nats or more in 79 of 81, and the smallest margin over it anywhere is 4.0 nats. The leak's size comes out with a bias of 0.0089 kg/s and an error of 0.029 kg/s against a mean posterior standard deviation of 0.036 kg/s, and 80 of the 81 intervals cover the truth.

The gap between 63 and 81 is not noise. Broken out by leak junction, the mean posterior on the true node is 1.000 at junctions 2, 3, 4 and 5 and 0.895 at junction 1 — and then 0.488, 0.490, 0.488 and 0.490 at junctions 6, 7, 8 and 9. Those four are the two structurally tied pairs of Section 22.2, and the posterior is doing exactly the right thing: it splits its mass evenly between two hypotheses that no reading can separate, and which of them is printed first is then a coin flip. Counting those as failures would be counting the posterior's honesty against it.

This is the same phenomenon Chapter 21 met with junctions 163, 164 and 166, and it is worth being precise about what it is. It is not an approximation error, and it does not shrink with more data: the two junctions have identical likelihood under every possible reading, so their evidence is identical to machine precision at any window length. A tie is not a failure to decide. It is a correct report that the question, as posed to these sensors, has two answers.

One more number says the probabilities are meaningful rather than merely ordered. The mean posterior mass on the true node over all 81 cases is 0.761, and the fraction of cases in which it actually ranks first is 0.778. A probability that says "I am right about three quarters of the time" and is right about three quarters of the time is calibrated.

22.5.2 What a window buys

Figure 22.3 gives the sweep. One snapshot puts 0.42 of the posterior mass on the true node and ranks it first in 14 of 27 cases. Twenty-four snapshots give 0.75 and 21 of 27. The leak's posterior standard deviation falls from 0.196 to 0.036 kg/s, and the evidence over the leak-free model grows from 5.6 to 277 nats.

The width falls 5.50-fold over a 24-fold increase in snapshots, where independent repeats of the same measurement would give $\sqrt{24} = 4.90$. The ratio to the $1/\sqrt{T}$ prediction is 0.90 at

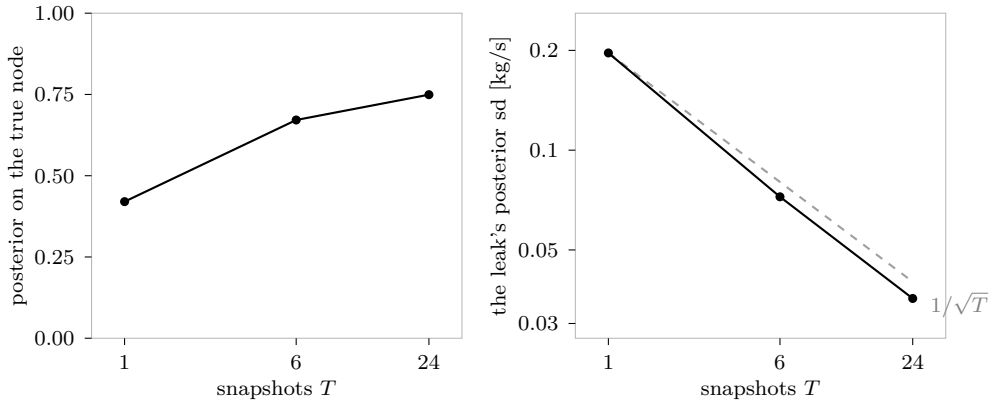


Figure 22.3: The window at $T = 1$, 6 and 24 snapshots, over 27 cases each. Left: the mean posterior mass on the true junction. Right: the leak's posterior standard deviation on log axes, against the dashed $1/\sqrt{T}$ line anchored at $T = 1$. The width falls slightly *faster* than $1/\sqrt{T}$ because the snapshots are not repeats of one measurement — each is taken under a different demand pattern.

$T = 6$ and 0.89 at $T = 24$ — about ten percent better than pooling. The reason is that the snapshots are not repeats. Each is taken under a different demand pattern, so the flows are differently distributed, the head drops load different pipes, and each snapshot interrogates a slightly different part of the network. More of the same measurement gives $1/\sqrt{T}$; more *different* measurements of the same constant give a little more.

For the hardest case — a leak at the junction fed directly by the reservoir, whose effect on the gauged heads is millimetric — long windows settle it outright: posterior 1.000 at $T = 96$ and again at $T = 288$, the latter over 8065 unknowns in 32 seconds for three solves.

22.5.3 What the leak competes with

Figure 22.4 is the chapter's answer to a question Chapter 21 raised and could not settle. There, a 5 kg/s leak at a node whose demand prior was 5.9 kg/s wide was invisible, and the explanation offered was that the leak had been absorbed into the demand. Here that explanation is measured directly, by widening the prior and watching.

At a prior of half a percent the true hypothesis beats the leak-free model by 265 nats and detection holds in all nine leaks. At two percent it is 54 nats and eight of nine. At five percent, 9.7 nats and three of nine. At ten percent, 1.4 nats and one of nine. Widening the prior twenty-fold costs a factor of 188 in evidence and widens the leak's own posterior 6.5-fold.

Notice what does *not* change much: the true node is still ranked first in seven of nine at every width up to five percent. The ranking survives long after the detection does. What the wide prior destroys is the ability to say that anything is wrong at all — and an operator who cannot pass that test has no reason to look at the ranking underneath it.

> **Principle 22.2.** *The information that separates a fault from an uncertain input is not in the size of the fault's effect but in the structure the fault is constrained to have. Widen the input's prior until it can mimic that structure and the fault disappears, however large it is.*

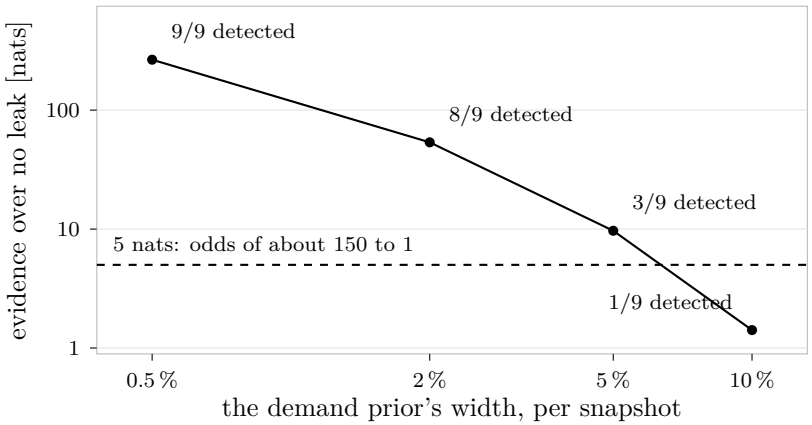


Figure 22.4: The mean evidence for the true leak over the leak-free model, as the per-snapshot demand prior is widened from half a percent to ten. The annotations count, out of nine leaks, how many the leak-free model is excluded for by at least five nats. The leak does not become harder to *find*; it becomes harder to *distinguish from a demand*, and past about five percent there is nothing left to distinguish.

Table 22.1: Three treatments of a Hazen–Williams roughness set ten percent too low, over the nine T-Example leaks. "Bias" is the mean error in the estimated leak, against a true leak of about 0.5 kg/s. The middle row is not a small degradation; it is a confident, precise, wrong answer.

the model's roughness	bias [kg/s]	top of 9	p(true) mean	evidence [nats]
true roughness	0.002957	7	0.775	265
10 % low, left fixed	−5.478	1	0.111	11,523
10 % low, calibrated on a leak-free day	−0.001148	7	0.768	261

22.5.4 A wrong roughness becomes a leak

Table 22.1 is the most important result in the chapter, and it is a failure.

Give the model the true roughness and the leak estimate is unbiased to 0.003 kg/s, with the true node first in seven of nine cases and 265 nats of evidence. Now set the roughness ten percent low — $C = 126$ where the truth is 140 — and leave it there. The model now believes each pipe needs less flow than it really does to produce the head drops it observes, and the deficit against the metered supply has to go somewhere. It goes to the leak. The estimated leak becomes −5.478 kg/s: eleven times the real leak, with the opposite sign. The true node is ranked first in one case of nine.

And the evidence over the leak-free model is 11,523 nats.

That number is the lesson. Eleven thousand nats is a certainty beyond any practical doubt, and it is attached to a leak that does not exist, of a size that is impossible, flowing the wrong way. Evidence compares hypotheses *within a model class*. It has nothing to say about a model class that does not contain the truth, and it will not warn you: a badly misspecified model does not produce an ambiguous answer, it produces a confident one.

The repair is cheap and it is the same device Modena needs anyway. Calibrate the roughnesses

Table 22.2: This chapter beside the four methods the benchmark's authors report, on the same 32 scenarios. Distances are from the true leak to the best-ranked candidate, and to the average of the five best, by shortest path through the network. The last row is the head estimate at the 252 junctions that carry no gauge.

	this chapter	FGLL	UKF-AW-GSI	GSI	GHR-S
best candidate [km]	**0.48**	0.72	0.69	0.93	1.47
best candidate [pipes]	**2.03**	3.34	3.28	4.38	7.37
average of five [km]	**0.55**	0.78	0.80	0.93	1.44
average of five [pipes]	**2.39**	3.60	3.81	4.37	7.12
head rmse, sensors only [m]	**0.38**	3.29	1.41	2.10	4.41

once on a day believed to be leak-free, with the pipe roughnesses as shared unknowns under a ten-percent prior, then hold them. Bias returns to -0.001 kg/s, the true node to seven of nine, the evidence to 261 nats. The calibration does not recover the individual roughnesses — three sensors cannot identify ten pipes, and the fitted values range from 123 to 143 around a true 140. It recovers the *combination* the leak search depends on, which is all that was needed.

Principle 22.3. *A misspecified model does not report its own misspecification. Evidence ranks hypotheses inside the class it is given; if the truth is outside that class, the ranking can be arbitrarily confident and arbitrarily wrong. The guard is not a better score but a separate observation — here, a day on which the fault is known to be absent.*

22.5.5 Modena: screen, then confirm

Calibration first. The nominal model, at $C = 130$ everywhere, misses the leak-free day by 13.6 cm rms and 43.1 cm at worst — the benchmark's authors put one percent noise on roughness and diameters in the model that generated it. Calibrated on six leak-free snapshots over 5309 unknowns, it misses by 3.3 cm at the junctions and 4.6 mm at the sensors, with the fitted C at 130.07 ± 0.45.

Then the screen. Across the 32 scenarios it puts the true junction in the twenty-five-name shortlist in 31, at a median rank of 2.5 out of 268. One step of Gauss–Newton against a cached factorization, applied to every junction in the network, ranks the right one second or third on average.

Then the exact evidence on those twenty-five. It puts the true junction first in 13 of 32, and excludes the leak-free model by five nats in all 32.

Table 22.2 and Figure 22.5 give the comparison. The best candidate is 0.48 ± 0.59 km from the true leak on average, 2.03 pipes, exactly right in 13 scenarios and within one pipe in 17; the average of the five best is 0.55 km. The published figures on the same scenarios are 0.72 and 0.78 km for the benchmark's own method and 0.69 and 0.80 for the strongest of the four. The head estimate at the 252 ungauged junctions is 0.38 m rms against 1.41 to 4.41 m for the published estimators — which is less a contest than a difference in kind, since those estimators interpolate between sensors and this one solves the network's physics at every node.

The runtime is 214 seconds per scenario, against 61 for the benchmark's own method and 936 for the strongest published one. That is the honest middle: faster than the best of the published methods and three times slower than the fastest.

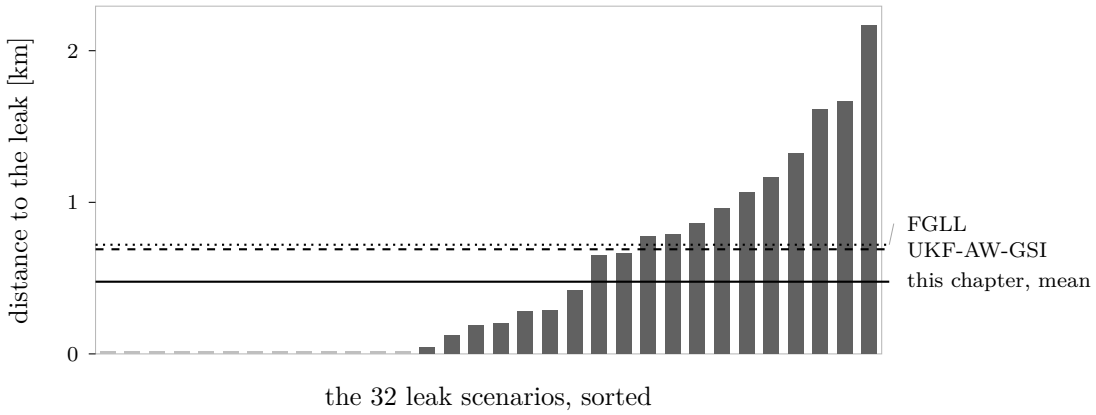


Figure 22.5: Every scenario, sorted by how far the best candidate falls from the true leak. The thirteen pale bars at the left are exact hits. The distribution is the result rather than the mean: a method that is exactly right thirteen times out of thirty-two and two kilometres wrong twice has a different operational character from one that is always half a kilometre off, even at the same average.

22.5.6 Cost

Every number above costs 3274 engine solves — 3238 most-probable-state solves, 36 Newton solves and 32 screenings — and just over two hours, of which the Modena localization is 6852 seconds.

The arithmetic worth keeping is per scenario. One leak-free solve, one screening over all 268 junctions, and 25 exact hypotheses: 26 solves where the method of Chapter 21 would have needed 269. The screen is what makes the difference, and it costs one sparse solve per junction against a factorization that the leak-free solve had already produced.

22.6 What surprised

Eleven thousand nats of confidence in a leak that is not there. The expectation going in was that a ten-percent roughness error would degrade the leak search — widen the posterior, blur the ranking, lose a few scenarios. It does not degrade it. It replaces the answer with a different one and reports the replacement with overwhelming certainty: a -5.478 kg/s leak, eleven times the true size and pointing the wrong way, at 11,523 nats over the leak-free model. Every internal diagnostic the chapter has is content. The misfit is fine, because the model can fit the data — with a leak that absorbs the roughness error. Nothing short of an observation from outside the model class catches it.

This is the sharpest available statement of what evidence is for. It ranks what you gave it. Chapter 20 used the misfit test as the guard that runs before the estimate; this chapter shows the misfit test passing on a model that is wrong in the one way it cannot see.

The screen is nearly free and nearly sufficient; the exact stage is where the difficulty lives. A single Gauss–Newton step per junction, against a factorization already computed, puts the true node at median rank 2.5 of 268 and inside a twenty-five-name shortlist in 31 of 32 scenarios. The expensive exact evidence on those twenty-five then promotes it to first in only

13. So the cheap stage does almost all of the useful work, and the expensive stage — which is exact, correct, and doing precisely what it should — still leaves the answer ambiguous more often than not. The remaining difficulty is not search and not approximation. It is that twenty gauges over two hundred and sixty-eight junctions do not determine the answer, and no amount of computation supplies information the sensors did not collect.

A window beats pooling. Twenty-four snapshots narrow the leak by 5.50-fold where $\sqrt{24} = 4.90$ was expected. The excess is small and it is structural: snapshots taken under different demand patterns are different experiments on the same constant, not repetitions of one. It is a mild result, but it points at something a designer can use — when choosing which six of a day's snapshots to carry, choose the six that differ most, not the six that are cleanest.

A tie is an answer. Eighteen of the 81 T-Example cases put the true node second, and every one of them is a structural tie at exactly equal evidence. Reported as recall, the method scores 78 percent. Reported honestly, it is right in all 81 and says so in the only way available: two junctions, half the mass each. Chapter 21 found the same thing on three junctions of Net3. The pattern is now established enough to state as a design rule — a localization method that always names one junction is concealing something the network's own topology guarantees.

22.7 Reproduction

The study takes about two hours, almost all of it in the Modena localization:

```
python docs/terra_graph_engine/case_studies/water_networks/\
       leak_localization_modena/run.py
```

It writes `results.json` and `timings.json` beside itself. The chapter's numbers and figures are then reproduced from those files alone:

```
cd docs/books/hydraulic && python scripts/w2_modena.py
```

which prints every value quoted above — including the ratios to $1/\sqrt{T}$ of Section 22.5.2 and the per-scenario arithmetic of Section 22.5.6, both derived from the recorded values — and rewrites the six fragments in `figures/`. A representative line of its output:

```
   T   p(true)     top1       nats   leak sd  1/sqrt(T)   ratio
  24    0.7494  21/27       276.64   0.03570    0.04009   0.890
```

The benchmark data ship with the study under the ISC license; no external solver is needed.

Notes

Both networks, their sensor sets, their scenarios and the four published comparison figures are Irofti et al. [35]'s, whose factor-graph formulation of leak localization over a window of snapshots is the direct precedent for this chapter; the comparison in Table 22.2 is against their reported means on their scenarios. The earlier factor-graph treatment of water-network state under damage is Han et al. [33], and the sparse least-squares machinery underneath both is Dellaert

and Kaess [17]. The Rao score test is standard statistics; its use here as a one-step screen against a cached factorization, and the observation that it over-scores on metered junctions because the one-step approximation does not see the meter's response, are worked out in Section 22.4. Evidence, the Occam factor and the Laplace approximation are Chapters 9 and 6.

What is this chapter's own is the measurement of Section 22.5.4 — that a ten-percent roughness error produces a leak estimate eleven times too large with the wrong sign at 11,523 nats of evidence, and that one calibration day repairs it — and the pairing in Section 22.5.3 of the demand-prior sweep with the unexplained misses of Chapter 21, which turns an after-the-fact explanation there into a measured curve here. The observation that a window narrows the estimate slightly faster than $1/\sqrt{T}$, because snapshots under different demand patterns are different experiments, appears to be new; it is small, and the chapter does not lean on it. Chapter 23 takes the same machinery to a network where the leak-free day is not available, because the utility does not know which days are leak-free.

Chapter 23

Held to what a utility knows

Numbers in this chapter are produced by `scripts/w3_battledim.py` from the study's committed results file. Nothing is retyped.

23.1 The question

Each chapter of this part has taken away one of the modeller's advantages.

Chapter 20 had a closed form to check against. Chapter 21 gave that up but kept a simulator that could produce the truth for every draw, so the error bars could be tested. Chapter 22 gave that up too, and leaned instead on a day known to be leak-free, which is what rescued it from the roughness error of Section 22.5.4.

This chapter has none of them. On this network the utility does not know which days are leak-free, because leaks run for months, several overlap, and some are repaired without anyone filing a report. There is no closed form, no per-draw truth, and no clean calibration day. What there is instead is a published benchmark with an official economic scorer, a ground truth that is revealed only afterwards, and six teams whose scores are on the record.

Two questions follow, and the whole chapter turns on the fact that their answers are independent.

- **What does it score?** Calibrate on the historical year, freeze every setting, then walk the evaluation year forward one night at a time without ever looking ahead — and submit to the benchmark's own scorer.
- **Are the probabilities it states honest?** Every report carries a credible set of pipes and an interval on the leak's size. Those are falsifiable, and the revealed truth falsifies them.

The published teams competed with the evaluation year's data in hand. This method does not, which makes the score comparison a generous reading in their favour and the honest one to make.

> **Principle 23.1.** *A method that is evaluated on data it has already seen is being asked an easier question than the one an operator faces. The discipline that makes a benchmark result mean something is fixing every setting before the evaluation period is read, and then not touching them.*

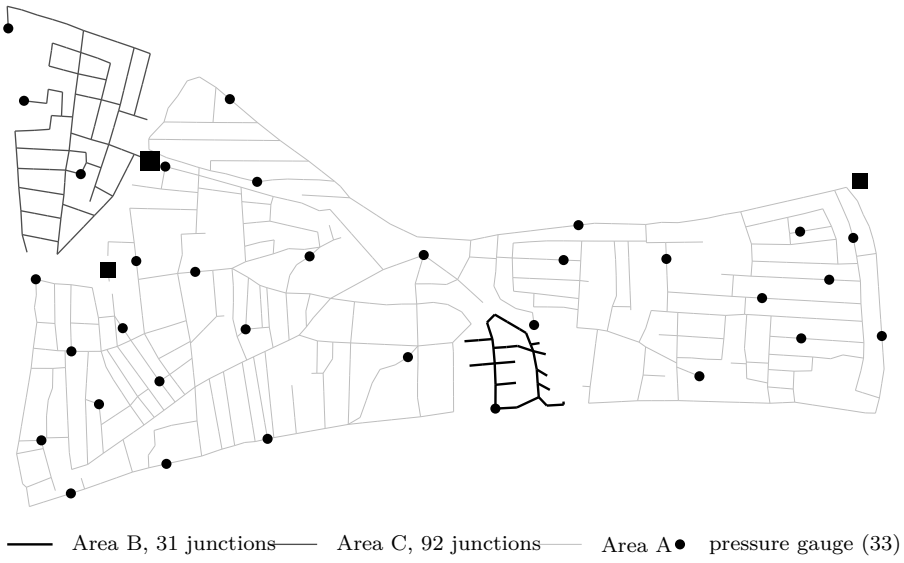


Figure 23.1: L-Town: 782 junctions on 905 pipes, fed by two reservoirs, a tank and one pump. Area C is fed from the tank and Area B through a pressure-reducing valve, which is what makes two separate nightly flow balances possible; Area A is everything else. Thirty-three pressure gauges serve the whole network.

23.2 The system

Figure 23.1 draws it. L-Town has 782 junctions on 905 pipes, two reservoirs, one tank, one pump and three valves. Two of its districts can be metered separately: Area C, the 92 junctions fed from the tank, and Area B, the 31 fed through a pressure-reducing valve. That division is what makes *two* nightly flow balances possible rather than one, and the chapter leans on it throughout.

The instrumentation is 33 pressure gauges, three flow meters and 82 demand meters, on two years of five-minute SCADA. Twenty-three leaks appear in the evaluation year, some starting during it and some already running when it begins.

23.3 The two layers

Figure 23.2 is the architecture. It is worth naming the relationship to the previous chapter: there, a cheap score test shortlisted *junctions* so that exact evidence ran on twenty-five instead of 268. Here a cheap flow balance shortlists *nights* so that the hydraulic model runs on a handful instead of 365. The pattern is the same and it is the only reason a year of five-minute data is tractable at all.

23.4 Method

Calibration, on the historical year. The nominal network model does not reproduce the recorded pressures. Pipe conductances are therefore scaled by one factor per diameter group — three groups — and the tank's cross-sectional area is made an unknown, all shared across a joint model of 24 snapshots, 59,325 unknowns in one solve. Every setting the detector uses is fixed

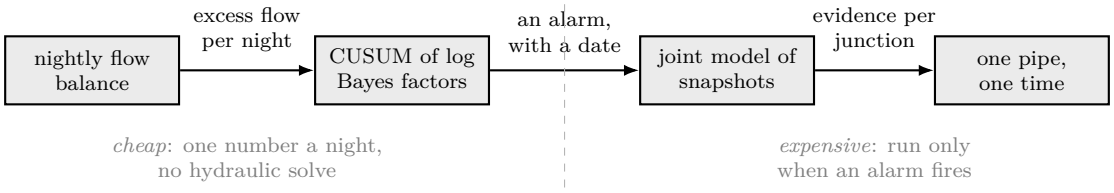


Figure 23.2: The two layers. A flow balance reduces a night to one number and costs no hydraulic solve at all, so it can run every night of the year. It decides *when*. Only when it raises an alarm does the expensive layer run — a joint hydraulic model of snapshots before and after the alarm, ranking one leak hypothesis per junction — which decides *where*. This is the screen-then-confirm structure of Chapter 22 applied along the time axis instead of across the network.

here, on the historical year only.

The nightly balance. For each of the two metered districts, the water entering at night minus the metered consumption is an estimate of what is being lost. Its noise is not white: fitted over 344 nights of the historical year, Area AB has a standard deviation of 1.58 m^3/h with an autocorrelation of 0.90 from one night to the next, and Area C 0.91 m^3/h with an autocorrelation of 0.11. Ignoring that correlation would make a run of ordinary nights look like a trend.

Detection. Each night contributes a log Bayes factor between two hypotheses — no new leak, against a new leak of a reference size of 20 m^3/h — and those are accumulated in a CUSUM, which raises an alarm when the running sum exceeds a threshold. Chapter 11 derives the accumulation; the threshold and the slack parameter are chosen on the historical year and then frozen.

Localization. An alarm names a date, not a place. The expensive layer then builds a joint hydraulic model of snapshots from before and after the alarm, adds one leak hypothesis per junction, and ranks them by evidence, exactly as Chapter 22 does. The report submitted to the scorer is the single best pipe, at the alarm's time.

The second evaluation. A leak that is repaired without a report stays in the detector's bookkeeping, which keeps subtracting it, so the balance carries a permanent excess that masks whatever comes next. A test for the *disappearance* of a known leak is therefore added, with its thresholds chosen on the historical year, and the whole evaluation run again. That second run is not blind — the failure it addresses was discovered by looking at the first run's behaviour — and Section 23.5.5 reports it as such.

23.5 Results

23.5.1 What calibration buys, and what it does not

Table 23.1 gives it. The nominal model misses the recorded pressures by 2.70 cm rms; calibrated, by 1.15 cm. The three conductance factors come out at 0.862 ± 0.013 for pipes up to 100 mm,

Table 23.1: The network model before and after calibration on the historical year. Three conductance factors and one tank area, shared across 24 snapshots, in a single solve over 59,325 unknowns. The worst-case misfit barely moves: the calibration fixes the systematic error and leaves the local one.

	nominal model	calibrated
pressure misfit, rms [cm]	2.70	1.15
pressure misfit, worst [cm]	10.92	9.73
conductance factor, pipes up to 100 mm	1	0.862 ± 0.013
conductance factor, 150 to 160 mm	1	1.066 ± 0.013
conductance factor, 200 mm and above	1	0.923 ± 0.004
tank diameter [m]	16.00	15.57

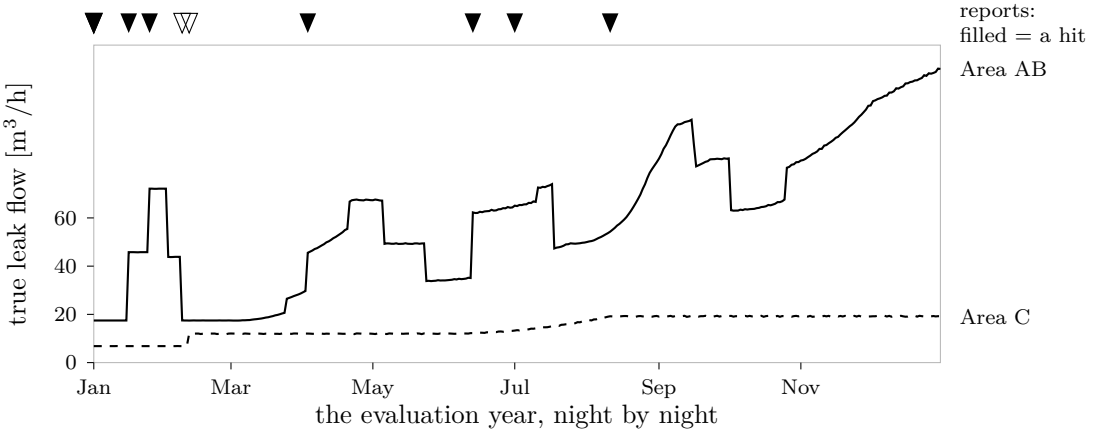


Figure 23.3: The true leak flow in each metered district, night by night through the evaluation year, with the eleven dated reports marked above: filled for a hit, open for a false positive. The burden in Area AB rises from 17.5 to 121.8 m^3/h across the year. The method is looking for *changes*, and the year's dominant feature is an accumulation.

1.066 ± 0.013 for 150 to 160 mm and 0.923 ± 0.004 above 200 mm, and the tank's diameter at 15.57 m against a nominal 16.0. No pipe changes hydraulic regime during the fit.

The row worth pausing on is the last of the misfit pair. The *worst* error falls only from 10.92 to 9.73 cm. Calibration has removed a systematic bias shared across the network and has not touched whatever is wrong at one particular place. That residue is not noise, and Section 23.5.4 is where it comes back.

23.5.2 The year, read forward

Figure 23.3 is what the year actually looked like, and it is worth studying before any score. On the first night of the year Area AB is already losing 17.5 m^3/h and Area C 6.8. On the last night they are losing 121.8 and 19.2. The network ends the year losing roughly seven times what it lost at the start.

This is a network that is being steadily lost, not one that suffers isolated events. Twenty-three leaks appear; seven are found.

Table 23.2: The benchmark's own evaluation. The six published teams competed with the evaluation year's data in hand; this method processed it causally with every setting frozen on the historical year. "Without the standing-leak submissions" removes the reports filed at the first instant of the year for leaks that were already running, and "with the closure test" is the second, not blind evaluation of Section 23.5.5.

submission	true positives	false positives	score
this method	7 of 23	4	EUR 128,805
without the day-one submissions	7 of 23	2	EUR 113,061
with the closure test (not blind)	7 of 23	5	EUR 111,561
Tongji	56.5 %	3	EUR 264,873
Under Pressure	65.2 %	4	EUR 260,562
IRI	43.5 %	1	EUR 210,772
Leakbusters	47.8 %	7	EUR 195,490
Tsinghua	47.8 %	5	EUR 167,981
UNIFE	43.5 %	4	EUR 127,626
a perfect submission	23 of 23	0	EUR 523,154

23.5.3 The official score

Table 23.2 gives it. The method finds 7 of the 23 leaks with 4 false positives and scores EUR 128,805, which is 24.6 percent of what a perfect submission would earn and above one of the six published teams. Its true-positive rate of 30.4 percent is below the published range of 43.5 to 65.2.

That is the headline, and it should be read with one caveat that says as much about the benchmark as about the method. Three of the reports are filed at the first instant of the year, for leaks that were already running when it began. One of them is a hit — a leak running since the previous January, reported 160 m from its true pipe — and it is worth EUR 46,805, which is 36.3 percent of the entire score. The scorer pays for the leak's whole remaining life, so naming a long-standing leak once, on day one, outearns catching a new one promptly. Drop those three reports and the score falls to EUR 113,061, with two false positives instead of four.

Neither number is wrong. They measure different things: one is what the benchmark pays, the other is what the method contributed over the year.

23.5.4 Are the stated probabilities honest?

They are not, and this is the chapter's real result.

Nineteen alarms were raised while a leak was running. Each carried a 90 % credible set of pipes. Four of them contained the true pipe: 21.1 percent, with a 95 % confidence interval of 7.5 to 41.9 percent, which excludes 90 by a wide margin. A set that claims 90 percent and delivers 21 is short by a factor of 4.3. The intervals on the leak's *size* do worse: two of nineteen cover, 10.5 percent. The stated probability of lying within 300 m of the truth scores 0.293 on the Brier scale, where zero is perfect and a quarter is what you get by saying "one half" every time.

Figure 23.4 shows the sets are not all the same width. The median holds five pipes; the narrowest holds two; one holds 621. So the method is not uniformly vague — it is confidently wrong on most alarms and occasionally, correctly, uncertain.

The cause is visible in Table 23.1. The calibrated model still misses by up to 9.7 cm somewhere

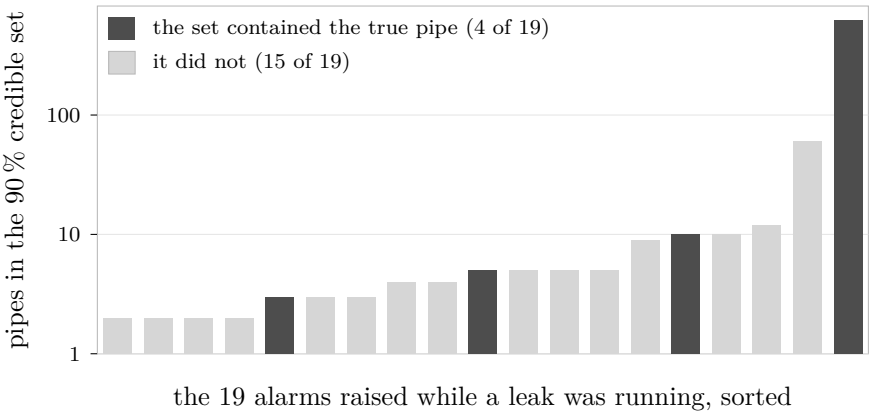


Figure 23.4: Every alarm raised while a leak was actually running, sorted by how many pipes its 90 % credible set contained. Dark bars are the sets that contained the true pipe. Four of nineteen did. The sets are not uniformly wide — the median holds five pipes and one holds 621 — so the failure is not simply that the method is vague.

in the network, and the leak model's noise budget treats its residuals as independent measurement error of a size fitted globally. A structured, spatially correlated model error is being scored as though it were white noise, so the evidence differences between neighbouring hypotheses come out far larger than they deserve. The ranking survives this, because the error is shared by all the hypotheses being compared. The *width* does not.

> **Principle 23.2.** *A ranking and an uncertainty can fail independently. Model error that is common to every hypothesis cancels in their comparison and does not cancel in the width of the posterior, so a method can order its candidates usefully while stating a confidence that is wrong by a factor of four.*

The operational reading is sharp. Reported as a ranked list, this method is worth money: it earns a benchmark score in the range of a competing team. The same output, reported as "we are 90 percent sure the leak is in this set of five pipes," would send crews to the wrong street four times in five, and nothing in the method's own diagnostics says so. Only the revealed truth does.

23.5.5 The second evaluation, which is worse

A leak repaired without a report is subtracted forever. The detector keeps crediting the network with a loss that has stopped, so the balance sits permanently high and a genuinely new leak has to clear that offset before it registers. The repair log for the year holds ten repairs.

The remedy is a test for the disappearance of a known leak, fired when the balance drops by about the size the known leak was carrying. Its thresholds were chosen on the historical year. Run over the evaluation year it fires nine closures, and it cuts the alarm count from 40 to 28.

It also scores *worse*. EUR 111,561 against EUR 128,805 — a loss of EUR 17,243, or 13.4 percent — with the same seven hits and one more false positive. Its credible sets cover three of seventeen rather than four of nineteen.

The reason is that the test matches drops by *size*. When two leaks in a district are of similar magnitude, a drop of the right size is consistent with either of them ending, and the test has

nothing else to go on. It closes the wrong one, the bookkeeping goes on subtracting a leak that is still running, and the detector is no better off than before — sometimes worse, because it has now also stopped subtracting one that really is there.

This is reported rather than repaired because the failure is informative. The test uses a real signal, it was tuned honestly on data the evaluation year never saw, and it makes things worse. A component being individually correct does not make a pipeline better.

23.5.6 Cost

Calibrating on the historical year and running both evaluations of the evaluation year takes about two hours. The hydraulic calibration itself is 59,325 unknowns in 32 seconds; almost all of the remaining time is the night-by-night walk through two years, and the localization solves that fire at each alarm.

23.6 What surprised

The score is respectable and the probabilities are not, and these are separate facts. EUR 128,805 places the method among the published field, above one of the six teams that had the answer year in hand. On the same run, its 90 % credible sets cover 21 percent of the time. Nothing in the benchmark's scoring can see the second failure, because the scorer counts hits and distances and never asks the method how sure it was. A method's ranking and a method's confidence are separately falsifiable, and a competition that scores only the first will not tell you about the second. Chapter 20 found a posterior twenty percent too narrow and a companion heating study one five times too wide; this is the same class of failure at the scale where it would actually dispatch a crew.

Adding a correct component made the system worse. The closure test detects a real thing, using a real signal, with thresholds chosen without touching the evaluation year. It costs EUR 17,243 and adds a false positive. The failure is compositional rather than local: a size match is sufficient to detect that *a* leak ended and insufficient to say *which*, and in a district with two similar leaks that ambiguity propagates into the bookkeeping that every later night depends on. The instinct to fix a known weakness by adding a detector for it is exactly what produced this, and the only thing that caught it was re-running the whole year and scoring it again.

The year's dominant feature is the one the detector cannot see. Area AB ends the year losing seven times what it lost at the start. A CUSUM on a nightly balance is a change detector: it is built to notice a step and it has no term for a slow accumulation. The method found seven of twenty-three leaks, and the seventh of the water that went missing over the year is not mostly attributable to the sixteen it missed individually — it is the accumulated standing burden that no single alarm corresponds to. A detector tuned to answer "did something change last night?" answers it well and leaves "how much are we losing, and is it growing?" to a different instrument.

23.7 Reproduction

The study takes about two hours:

```
python docs/terra_graph_engine/case_studies/water_networks/\
       battledim_l_town/run.py
```

It writes `results.json` and `timings.json` beside itself. The chapter's numbers and figures are then reproduced from those files alone:

```
cd docs/books/hydraulic && python scripts/w3_battledim.py
```

which prints every value quoted above — including the standing-leak share of Section 23.5.3 and the coverage shortfall of Section 23.5.4, both derived from the recorded rows — and rewrites the five fragments in `figures/`. A representative line of its output:

```
  ... sets that contained the true pipe   4 of 19 (21.1 %, 7.5-41.9 %)
  shortfall: a 90 % set covering 21.1 % is short by a factor of 4.3
```

The benchmark data and its official scorer ship with the study; no external solver is needed.

Notes

The network, the two years of SCADA, the leak ground truth, the economic scorer and the six published teams' figures are the BattLeDIM benchmark's, whose report defines the scoring rule and tabulates the competition results; the comparison in Table 23.2 quotes those figures and the perfect-submission total from it. The factor-graph formulation of the localization layer follows Irofti et al. [35] and Han et al. [33], as in Chapter 22, on the sparse least-squares machinery of Dellaert and Kaess [17]. The cumulative sum as a change detector, and its formulation as an accumulation of log likelihood ratios, are classical; Chapter 11 derives the accumulation and Chapter 9 the evidence.

What is this chapter's own is the causal protocol — every setting fixed on the historical year and the evaluation year walked forward without lookahead — together with the calibration audit of Section 23.5.4: that the method's ranking survives a structured model error which its width does not, so a benchmark score in the published range coexists with 90 % credible sets that cover 21 percent of the time. The negative result of Section 23.5.5, that adding a correctly-tuned closure test lowers the score, is also the chapter's, and is reported rather than repaired. A companion parameter study leaves synthetic and benchmark data behind for a laboratory rig, where the unknowns are the network's parameters rather than its faults.

Appendix A

Notation

This appendix collects the book's notation. Section A.1 states the conventions that hold throughout. Section A.2 lists the symbols by subject, each with its meaning and the place that introduces it. Section A.3 lists the letters that carry more than one meaning, and where each meaning applies. A symbol used in only one example, one proof or one closure is defined where it is used and is not listed.

A.1 Conventions

Type. Scalars are italic: z_k, σ, G. Vectors are bold lowercase, $\mathbf{z}$, $\mathbf{h}$, and matrices bold capitals, $\mathbf{J}$, $\mathbf{K}$; Greek vectors and matrices are bold as well, $\boldsymbol{\mu}$, $\boldsymbol{\Sigma}$, $\boldsymbol{\Lambda}$. A vector is a column, $\mathbf{z}^\mathsf{T}$ is its transpose, and $(\mathbf{a};\mathbf{b})$ stacks two columns into one. $\mathbf{I}$ is the identity, $\mathbf{0}$ and $\mathbf{1}$ are the vectors of zeros and ones, and $\mathbf{e}_k$ is the kth unit vector. Calligraphic capitals are sets or objects built from sets: $\mathcal{N}$, $\mathcal{E}$, the manifold $\mathcal{M}$, the regime $\mathcal{R}_b$. $\mathrm{diag}(\mathbf{w})$ is the diagonal matrix with $\mathbf{w}$ on its diagonal, and tr, det, $\mathbb{E}$, Var and Cov are the trace, determinant, expectation, variance and covariance.

The unknowns and the rows. The unknowns of a model are stacked into one vector $\mathbf{z} \in \mathbb{R}^n$, flows first and potentials after: $\mathbf{z} = (\mathbf{m};\mathbf{h})$ for the pumping chain of Example 1.1, and pipe flows before junction heads for the water triangle of Example 1.2. When a later chapter frees an input of an example, the input is appended to the example's $\mathbf{z}$, so the order a reader has learned is never changed. Flows first is an order for listing, not for elimination; Chapter 5 chooses elimination orders on other grounds. The residual $\mathbf{F}(\mathbf{z}) \in \mathbb{R}^m$ lists its rows in the order of §1.5: balances, edge closures, storage relations, freestanding closures, and boundary-condition rows. In an example the balance rows come first and the closure rows after, and the text says which is which.

Inputs and solved variables. The inputs $\mathbf{u}$ are the quantities a modeler places priors on; the solved variables $\mathbf{x}$ are the ones the physics then determines, through the solve map $\mathbf{x} = X(\mathbf{u})$ (§4.1). When the physics is linear the whole state is a linear image of the inputs, $\mathbf{z} = \mathbf{S}\mathbf{u}$ (§3.4).

Widths and weights. Every factor has a width σ, in the units of its residual, and a weight $w = 1/\sigma^2$; $\boldsymbol{\Omega}$ is the diagonal matrix of the weights of all factors (§6.1). A residual divided by its width is *standardized* (§3.3.1).

Gaussians. $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the Gaussian with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$, and $\mathcal{N}(\mathbf{z}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is its density at $\mathbf{z}$. The information form has precision $\boldsymbol{\Lambda} = \boldsymbol{\Sigma}^{-1}$ and information vector $\boldsymbol{\eta} = \boldsymbol{\Lambda}\boldsymbol{\mu}$ (equations (B.1) and (B.2)); a density written with $(\boldsymbol{\eta}, \boldsymbol{\Lambda})$ in place of $(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is marked as such where it appears. A covariance may be singular (Definition B.1).

Decorations. A prime marks the result of conditioning on readings: $\boldsymbol{\mu}'$ and $\boldsymbol{\Sigma}'$ are a posterior mean and covariance, and $\boldsymbol{\Sigma}''$ follows a second reading. A star marks a mode: $\mathbf{z}^\star$ is the most probable state (§6.2) and $\mathbf{z}_a^\star$ the mode of branch a. A bar marks either an imposed value, $\bar{z}_k$, or a sample mean, $\bar{G}_N$. In Chapter 16 a superscript counts time steps, $\mathbf{x}^n$, and the subscript $n+1 \mid n$ marks a prediction from the readings up to step n. Chapters 13 and 15 use the prime for other purposes, listed in Table A.7.

Drawings. In a factor graph, circles are variables and squares are factors; a dark square is a prior and an open square is a sensor (Figure 3.1).

Local symbols. The cooling loop of Example 1.3, the entries of Appendix D (§D.1) and the case studies of Part VI define symbols of their own where they use them. The case studies keep the book's symbols where the meaning is the same.

A.2 Symbols

Tables A.1 to A.6 give the symbols by subject, in the order the book introduces the subjects. The last column names the first place a symbol is defined; many are used throughout.

Table A.1: Graphs, sets and indices.

symbol	meaning	introduced
$\mathcal{N}$, $\mathcal{E}$, $\mathcal{D}$	nodes, edges, and conservation domains of a physical graph	§1.2
n, e, d	a node, an edge, a domain	equation (1.1)
$\mathcal{E}(n)$	the edges incident on node n	equation (1.1)
$A_{n,e}$, $\mathbf{A}$	incidence: $+1$ or -1 by the orientation of edge e at node n, 0 otherwise; one row per inside node	§1.2.3
$\mathbf{A}_b$	the incidence rows of the reservoirs	Example 1.1
reservoir, port	a node whose condition is imposed; a flow and potential pair at a boundary	§1.4, Table 1.3
S, ∂S	a set of inside nodes; the edges with one end in it	Proposition 1.1, equation (17.2)
N, E, S, Θ, C, R	counts of inside nodes, edge channels, storages, freed parameters, freestanding closures and boundary rows	Proposition 1.2
φ_f, $\mathbf{z}_f$	a factor and its scope, the variables it reads	§2.5
$\mathrm{pa}(i)$	the parents of variable i in a directed graph	equation (2.1)
$x_1 \perp x_3 \mid x_2$	x_1 and x_3 are independent given x_2	equation (5.1)
A, B, S	sets of variables; S separates A from B	Definition 5.1
$\mathcal{C}$	the set of cut edges	Proposition 5.3
width, treewidth	the largest scope an elimination order creates, less one; its minimum over orders	Definition 5.2

Table A.1, continued

symbol	meaning	introduced
R_A	the variables of region A	§5.7
r, I_r, S	a region, its interior variables, the port set	§8.4
n_r, k_r, w_r	the interior variables, ports and elimination width of region r	§18.1

Table A.2: The physical model.

symbol	meaning	introduced
$X_{n,d}$	the extensive state of domain d at node n	equation (1.1)
$P_{d,e}$	the flow of domain d along edge e; written Q for heat and F for DC power in the examples	equation (1.1)
$G_{n,d}$, $L_{n,d}$	a source and a loss at a node	equation (1.1)
$r(\mathbf{z}_{\text{loc}}; \theta) = 0$, θ	a closure in residual form; its parameters	§1.3
$\mathbf{z} \in \mathbb{R}^n$	the unknowns, flows first	§1.5
$\bar{z}_k$	a value imposed at a reservoir	§1.5
$\mathbf{F}(\mathbf{z}) = \mathbf{0}$, m	the residual system; its number of rows	equation (1.7)
$\mathbf{J} = \partial\mathbf{F}/\partial\mathbf{z}$	the Jacobian	§1.5
$\mathbf{m}$, $\mathbf{h}$, $\mathbf{s}$	the pipe flows, the junction heads, and the sources at the inside nodes	Example 1.1
k, k_2, h_a	the rising main's conductance; the throttling valve's conductance; the trunk main's head	Example 1.1
$\mathbf{K}$	the diagonal matrix of conductances or susceptances	equation (1.13)
$\mathbf{m}$, $\mathbf{h}$, k	the pipe flows and junction heads of the water loop; a conductance	Example 1.2
C	a storage capacity, in a storage row $M = C\,h$	§1.5, equation (16.3)
$\mathbf{u}$, $\mathbf{x}$	inputs; solved variables	§4.1
$X(\mathbf{u})$	the solve map from inputs to solved variables	equation (4.1)
$\mathcal{M}$	the constraint manifold, the states that satisfy the physics	equation (4.1)
$\mathbf{J}_x$, $\mathbf{J}_u$	the Jacobian's blocks for solved variables and for inputs	§4.1
$\mathbf{S}$	the linear map from inputs to unknowns, $\mathbf{z} = \mathbf{S}\mathbf{u}$	§3.4
$\boldsymbol{\lambda}$	the adjoint vector	§14.2
$\mathbf{F}(\mathbf{z}, \dot{\mathbf{z}}, t) = \mathbf{0}$	the differential-algebraic system	equation (16.2)

Table A.3: Probability and factors.

symbol	meaning	introduced
$p(\mathbf{z})$, $p(\mathbf{z}_1 \mid \mathbf{z}_2)$	a density; a conditional density	§3.1
$\mathbb{P}$, $\mathbb{E}$	probability; expectation	Chapter 3
Z	the normalizer of a density	§3.1
$\mathcal{N}(\mu, \sigma^2)$	the Gaussian with mean μ and variance σ^2	equation (3.1)
π_j, μ_j, σ_j	a prior factor on z_j; its mean and width	§3.2
φ_k, σ_k	a physics factor; its width	equation (3.3)

Table A.3, continued

symbol	meaning	introduced
y_i, z_{o_i}	a reading; the variable it observes	equation (3.5)
ν_i, σ_i	a sensor's noise; its accuracy, σ_y for a single reading	equation (3.5)
ℓ_i	the likelihood factor of reading i	equation (3.5)
a, q	the availability of a component, 0 or 1; its failure rate	equation (3.6)
$p(\mathbf{z} \mid \mathbf{y})$	the posterior	equation (3.7)
$\mathbf{y}$, $\mathbf{H}$, $\mathbf{R}$	the stacked readings, the sensor matrix, the noise covariance: $\mathbf{y} = \mathbf{H}\mathbf{z} + \boldsymbol{\nu}$	Proposition 3.2
$\mathbf{h}$	the sensor vector of one reading, $y = \mathbf{h}^\mathsf{T}\mathbf{z} + \nu$	Proposition 3.1
χ^2, $m_g - n$	the misfit; its degrees of freedom	Proposition 12.1
Z_a, C_a	the evidence of branch a; its objective	equations (7.1) and (7.2)
$\mathcal{R}_b$, $p_b(\mathbf{y})$	the region of input space where branch b holds; the branch's predictive density of the readings	Proposition 7.2
$\mathrm{KL}(q \,\|\, p)$	the Kullback–Leibler divergence	Proposition 9.2, equation (B.12)
L, S, k	the number of components that can fail; a failure set; its size	§15.1
$\mathbb{P}(S)$, q_i	the prior of a failure set; the failure probability of component i	§15.1, equation (15.1)

Table A.4: Linear-Gaussian inference.

symbol	meaning	introduced
$\boldsymbol{\mu}$, $\boldsymbol{\Sigma}$	a mean and a covariance	Proposition 3.1
$\boldsymbol{\Lambda} = \boldsymbol{\Sigma}^{-1}$, $\boldsymbol{\eta} = \boldsymbol{\Lambda}\boldsymbol{\mu}$	the precision; the information vector	Proposition 3.2
$\boldsymbol{\mu}'$, $\boldsymbol{\Sigma}'$, $\boldsymbol{\Lambda}'$, $\boldsymbol{\eta}'$	the same after conditioning on readings	equations (3.9) and (3.10)
e, s	the innovation $y - \mathbf{h}^\mathsf{T}\boldsymbol{\mu}$; its predicted variance. Indexed e_i, s_i by reading and e_n, s_n by time step	Proposition 3.1
$\mathbf{k}$, $\mathbf{s}$	the gain of one reading; the vector $\boldsymbol{\Sigma}\mathbf{h}$	equation (11.1)
$\mathbf{S}$, $\mathbf{K}$	the innovation covariance $\mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^\mathsf{T} + \mathbf{R}$ of several readings; their gain	equation (B.6)
w, $\boldsymbol{\Omega}$	a factor's weight $1/\sigma^2$; the diagonal matrix of weights	§6.1
$\mathbf{g}(\mathbf{z})$, m_g, $\mathbf{J}_g$	every residual stacked, physics, prior and sensor; its length; its Jacobian	equation (6.3)
$C(\mathbf{z})$	the cost, $\frac{1}{2}\mathbf{g}^\mathsf{T}\boldsymbol{\Omega}\mathbf{g}$, the negative log posterior up to a constant	equation (6.4)
$\mathbf{z}^\star$	the most probable state	§6.2
$\boldsymbol{\delta}$	a Gauss–Newton step	equation (6.6)
$\nabla^2 C$	the Hessian of the cost	equation (6.7)
$\mathbf{L}$	the Cholesky factor, $\boldsymbol{\Lambda} = \mathbf{L}\mathbf{L}^\mathsf{T}$	§6.3
$\boldsymbol{\xi}$	a vector of independent standard normals	Proposition 6.4
$\mathbf{M}/\mathbf{D}$	the Schur complement of the block $\mathbf{D}$ in $\mathbf{M}$	Lemma B.4
$m_{x\to f}$, $m_{f\to x}$	sum-product messages	equations (8.1) and (8.2)
$\mathbf{S}_r$, $\mathbf{h}_r$	the precision and information of region r's message on the ports	equation (8.5)
$\mathbf{X}$	the sensitivity of a region's interior to its ports	equation (8.6)
EIG	the expected information gain of a reading	equation (12.3)

Table A.4, continued

symbol	meaning	introduced
$\mathbf{F}$, $\mathbf{Q}$, $\mathbf{w}$	the transition matrix, process-noise covariance and process noise of a linear dynamic model	Proposition 16.3

Table A.5: Time.

symbol	meaning	introduced
t, t_n, Δt	time; the nth time point; the step	§16.2
X^n, $\mathbf{z}^{n+1}$	a superscript counts time steps	equation (16.4)
$\boldsymbol{\mu}_{n+1\|n}$, $\boldsymbol{\Sigma}_{n+1\|n}$	the prediction of step $n+1$ from readings to step n	equation (16.6)
$\log Z$	the evidence of a record of readings in time	equation (16.8)

Table A.6: Consequences and attribution.

symbol	meaning	introduced
$G(\mathbf{u})$	a scalar consequence of the physics as a function of the inputs	equation (10.1), §14.1
$C(S)$	the consequence of failure set S	§15.1
S_i	the first-order Sobol index of input i	§14.3
$L(\mathbf{u})$, $\mathbf{s}_j$	a cutting-plane surrogate below G; a subgradient of G at census point j	equation (14.3), Proposition 14.2
PTDF, LODF	power-transfer and line-outage distribution factors	equation (15.2)

A.3 Letters with more than one meaning

A book that spans physics, probability and computation runs out of letters. Table A.7 lists the letters that carry different meanings in different places, with the scope of each meaning. Within a chapter a letter has one meaning unless the chapter says otherwise.

Table A.7: Letters with more than one meaning, and the scope of each.

letter	meanings and where they apply
A, $\mathbf{A}$	the incidence (Chapter 1 on); a generic matrix in lemmas and in Appendix B; sets or regions A, B (Chapters 5, 8, 13, 15); the area in k_2
C	a storage capacity; a count in Proposition 1.2; the cost $C(\mathbf{z})$ of Chapter 6; the consequence $C(S)$ of Chapter 15; the census size in Chapter 14; the compression C_r of Chapter 18
e, $\mathbf{e}_k$	an edge (Chapter 1); the innovation (Chapters 3, 11, 16); the unit vector $\mathbf{e}_k$
F, $\mathbf{F}$	the residual $\mathbf{F}(\mathbf{z})$ throughout; the line flows F_e and $\mathbf{F}$ of DC networks; the transition matrix $\mathbf{F}$ of Proposition 16.3
G, $\mathbf{G}$	a source at a node, including the pumping chain's delivery G in every chapter that uses the chain; the consequence $G(\mathbf{u})$ of Chapters 10 and 14 and the case studies; the laminar conductances $\mathbf{G}$ of Example 1.3; the blocks $\mathbf{G}_x$, $\mathbf{G}_u$ of Chapter 13
$\mathbf{g}$	the stacked residual of every factor, Chapter 6 on

Table A.7, continued

letter	meanings and where they apply
h, $\mathbf{h}$	the sensor vector $\mathbf{h}$ (Chapter 3 on); a junction head h_n and the head vector $\mathbf{h}$ (Example 1.1 on); the region information $\mathbf{h}_r$ (Chapter 8); enthalpy in Appendix D; a step size in Chapters 10 and 14
H, $\mathbf{H}$	the sensor matrix; the Fisher information in Chapter 12; the entropy $H[p]$ of equation (B.11)
J, $\mathbf{J}$	the Jacobian throughout; a slope jump in Chapters 10 and 14; the interdiction objective $J(A)$ of Chapter 15
K, k	the conductance matrix $\mathbf{K}$ and a conductance k; the gain $\mathbf{k}$ of one reading and $\mathbf{K}$ of several; the size k of a failure set (Chapter 15); the ports k_r of a region (Chapters 8 and 18); the number of uncertain inputs K in Chapter 14
L, $\mathbf{L}$	a loss at a node; the Cholesky factor $\mathbf{L}$; the number of components L (Chapter 15); the surrogate $L(\mathbf{u})$ (Chapter 14)
P, $\mathbf{P}$	a flow $P_{d,e}$; a projector
Q, q	the process noise $\mathbf{Q}$ (Chapter 16); the unserved consequence Q of Chapter 15; a failure rate q; a variational density q (Chapter 9); a nodal source or draw q_n, negative for a demand (Example 1.2 on)
R, r	the noise covariance $\mathbf{R}$; a count in Proposition 1.2; a residual r; a region r (Chapter 8 on)
S, $\mathbf{S}$, s	a set of nodes (Chapter 1), of variables (Chapter 5), of ports (Chapter 8) or of failed components (Chapter 15); the map $\mathbf{z} = \mathbf{S}\mathbf{u}$; the innovation covariance $\mathbf{S}$ and variance s; the region message $\mathbf{S}_r$; the Sobol index S_i (Chapter 14); the sources $\mathbf{s}$ of Chapter 1; the vector $\mathbf{s} = \mathbf{\Sigma h}$ (Chapter 11); a subgradient $\mathbf{s}_j$ (Chapter 14)
T, $^\mathsf{T}$	a temperature; the transpose
w	a weight $1/\sigma^2$; the width of an elimination order and w_r of a region; process noise $\mathbf{w}$ (Chapter 16); a scenario weight w_s (Chapter 15)
X, x	the extensive state $X_{n,d}$; the solve map $X(\mathbf{u})$; the variables X_i and scopes X_f of a generic graphical model in Chapter 2; the interior sensitivity $\mathbf{X}$ (Chapter 8); the solved variables $\mathbf{x}$, and the state $\mathbf{x}^n$ of Chapter 16
Z	a normalizer; the evidence Z_a of a branch or hypothesis
θ	the parameters of a closure
ν	a sensor's noise
φ	a factor; a potential φ_i at a port (Chapters 5 and 17)
λ, ρ, ε	each is defined where it is used: $\boldsymbol{\lambda}$ the adjoint (Chapter 14), an eigenvalue, a scalar precision; ρ a correlation, a relative width, a spectral radius; ε a noise, a discarded weight (Proposition 9.3)
prime	a posterior (Chapters 3 and 11, Appendix B); a replaced mechanism or the intervened world (Definition 13.1); the post-outage flow F'_e and the reduced susceptance $\mathbf{B}'$ of the linear-network result of Proposition 15.2

Appendix B

Gaussian identities

This appendix collects the facts about Gaussian distributions that the book uses, each with a short proof. Most chapters need only a few of them, and the table in §B.8 says which chapter uses which. Where a chapter already proves an identity in the form it needs, the appendix states the identity and points to that proof. Every identity is checked numerically, on randomly drawn matrices, by the script `appB_gaussian_identities.py` in the book's `scripts` directory.

Throughout, $\mathbf{z} \in \mathbb{R}^n$, a covariance $\boldsymbol{\Sigma}$ is symmetric positive semidefinite, and a precision $\boldsymbol{\Lambda}$ is symmetric positive definite. Bold capitals are matrices, $\mathbf{I}$ is the identity, and tr and det are the trace and the determinant.

B.1 The two forms of a Gaussian

A Gaussian with mean $\boldsymbol{\mu}$ and positive definite covariance $\boldsymbol{\Sigma}$ has the density

$$\mathcal{N}(\mathbf{z}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) \;=\; (2\pi)^{-n/2} \, (\det \boldsymbol{\Sigma})^{-1/2} \exp\!\big(-\tfrac{1}{2}(\mathbf{z}-\boldsymbol{\mu})^{\mathsf{T}} \boldsymbol{\Sigma}^{-1} (\mathbf{z}-\boldsymbol{\mu})\big). \tag{B.1}$$

This is the *moment form*. Expanding the quadratic with $\boldsymbol{\Lambda} = \boldsymbol{\Sigma}^{-1}$ and $\boldsymbol{\eta} = \boldsymbol{\Lambda}\boldsymbol{\mu}$ gives the *information form* of Chapter 6,

$$\mathcal{N}(\mathbf{z}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) \;=\; \exp\!\big(-\tfrac{1}{2}\mathbf{z}^{\mathsf{T}}\boldsymbol{\Lambda}\mathbf{z} + \boldsymbol{\eta}^{\mathsf{T}}\mathbf{z} - \kappa\big), \qquad \kappa = \tfrac{1}{2}\boldsymbol{\eta}^{\mathsf{T}}\boldsymbol{\Lambda}^{-1}\boldsymbol{\eta} + \tfrac{n}{2}\log 2\pi - \tfrac{1}{2}\log\det\boldsymbol{\Lambda}. \tag{B.2}$$

The two carry the same information. The moment form makes marginals and predictions easy, and the information form makes products and conditioning easy, which is the division of labour §B.3 makes precise.

Lemma B.1 (The Gaussian integral)**.** *For $\boldsymbol{\Lambda}$ positive definite and any $\boldsymbol{\eta}$,*

$$\int_{\mathbb{R}^n} \exp\!\big(-\tfrac{1}{2}\mathbf{z}^{\mathsf{T}}\boldsymbol{\Lambda}\mathbf{z} + \boldsymbol{\eta}^{\mathsf{T}}\mathbf{z}\big)\,\mathrm{d}\mathbf{z} \;=\; (2\pi)^{n/2}\,(\det\boldsymbol{\Lambda})^{-1/2} \exp\!\big(\tfrac{1}{2}\boldsymbol{\eta}^{\mathsf{T}}\boldsymbol{\Lambda}^{-1}\boldsymbol{\eta}\big).$$

Proof. Complete the square: with $\mathbf{m} = \boldsymbol{\Lambda}^{-1}\boldsymbol{\eta}$ the exponent is $-\frac{1}{2}(\mathbf{z}-\mathbf{m})^{\mathsf{T}}\boldsymbol{\Lambda}(\mathbf{z}-\mathbf{m}) + \frac{1}{2}\boldsymbol{\eta}^{\mathsf{T}}\boldsymbol{\Lambda}^{-1}\boldsymbol{\eta}$. Factor $\boldsymbol{\Lambda} = \mathbf{L}\mathbf{L}^{\mathsf{T}}$ by Cholesky and substitute $\mathbf{w} = \mathbf{L}^{\mathsf{T}}(\mathbf{z}-\mathbf{m})$, so that $\mathrm{d}\mathbf{z} = \mathrm{d}\mathbf{w}/\det\mathbf{L}$ and the quadratic becomes $\mathbf{w}^{\mathsf{T}}\mathbf{w}$. The integral of $\exp(-\frac{1}{2}\mathbf{w}^{\mathsf{T}}\mathbf{w})$ is a product of n one-dimensional integrals, each $\sqrt{2\pi}$, and $\det\mathbf{L} = (\det\boldsymbol{\Lambda})^{1/2}$. □

The lemma is what normalizes equations (B.1) and (B.2), and it has a corollary used on almost every page of Part III: *a density whose logarithm is $-\frac{1}{2}\mathbf{z}^{\mathsf{T}}\boldsymbol{\Lambda}\mathbf{z} + \boldsymbol{\eta}^{\mathsf{T}}\mathbf{z}$ plus a constant is the*

Gaussian with precision $\mathbf{\Lambda}$ *and mean* $\mathbf{\Lambda}^{-1}\boldsymbol{\eta}$. To find a posterior, collect the quadratic and linear terms of its logarithm and read the two off.

A Gaussian need not have a density. When the physics is exact, as in the examples of Chapter 3, the unknown vector is a linear function of a few inputs, $\mathbf{z} = \mathbf{S}\mathbf{u}$, and its covariance $\mathbf{S}\mathbf{\Sigma}_u\mathbf{S}^\mathsf{T}$ is singular. The general definition goes through the moment generating function.

Definition B.1 (Gaussian vector)**.** A random vector $\mathbf{z}$ is Gaussian with mean $\boldsymbol{\mu}$ and covariance $\mathbf{\Sigma}$, written $\mathbf{z} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{\Sigma})$, if $\mathbb{E}[\exp(\mathbf{t}^\mathsf{T}\mathbf{z})] = \exp(\mathbf{t}^\mathsf{T}\boldsymbol{\mu} + \frac{1}{2}\mathbf{t}^\mathsf{T}\mathbf{\Sigma}\mathbf{t})$ for every $\mathbf{t} \in \mathbb{R}^n$. It has the density (B.1) exactly when $\mathbf{\Sigma}$ is positive definite.

For a positive definite $\mathbf{\Sigma}$ the definition agrees with the density: multiplying equation (B.2) by $\exp(\mathbf{t}^\mathsf{T}\mathbf{z})$ and integrating by Lemma B.1, with $\boldsymbol{\eta}$ replaced by $\boldsymbol{\eta} + \mathbf{t}$, gives the stated function. A moment generating function that is finite everywhere determines its distribution, so the definition names each Gaussian exactly once.

Lemma B.2 (Affine maps)**.** *If* $\mathbf{z} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{\Sigma})$ *and* $\mathbf{w} = \mathbf{A}\mathbf{z} + \mathbf{b}$ *for an* $m \times n$ *matrix* $\mathbf{A}$*, then* $\mathbf{w} \sim \mathcal{N}(\mathbf{A}\boldsymbol{\mu} + \mathbf{b},\ \mathbf{A}\mathbf{\Sigma}\mathbf{A}^\mathsf{T})$*. In particular every block of a Gaussian vector is Gaussian, with the corresponding blocks of* $\boldsymbol{\mu}$ *and* $\mathbf{\Sigma}$*.*

Proof. $\mathbb{E}[\exp(\mathbf{s}^\mathsf{T}\mathbf{w})] = \exp(\mathbf{s}^\mathsf{T}\mathbf{b})\,\mathbb{E}[\exp((\mathbf{A}^\mathsf{T}\mathbf{s})^\mathsf{T}\mathbf{z})] = \exp\big(\mathbf{s}^\mathsf{T}(\mathbf{A}\boldsymbol{\mu} + \mathbf{b}) + \frac{1}{2}\mathbf{s}^\mathsf{T}\mathbf{A}\mathbf{\Sigma}\mathbf{A}^\mathsf{T}\mathbf{s}\big)$. A block is the map $\mathbf{A} = (\mathbf{0}\ \ \mathbf{I})$. □

Lemma B.3 (Uncorrelated blocks are independent)**.** *If* $(\mathbf{a}, \mathbf{b})$ *is Gaussian and* $\operatorname{Cov}(\mathbf{a}, \mathbf{b}) = \mathbf{0}$*, then* $\mathbf{a}$ *and* $\mathbf{b}$ *are independent.*

Proof. With a zero cross-covariance the joint moment generating function is

$$\mathbb{E}\big[\exp(\mathbf{s}^\mathsf{T}\mathbf{a} + \mathbf{t}^\mathsf{T}\mathbf{b})\big] = \exp\big(\mathbf{s}^\mathsf{T}\boldsymbol{\mu}_a + \tfrac{1}{2}\mathbf{s}^\mathsf{T}\mathbf{\Sigma}_{aa}\mathbf{s}\big)\,\exp\big(\mathbf{t}^\mathsf{T}\boldsymbol{\mu}_b + \tfrac{1}{2}\mathbf{t}^\mathsf{T}\mathbf{\Sigma}_{bb}\mathbf{t}\big),$$

the product of the two marginal ones, which is the moment generating function of an independent pair. □

B.2 Block matrices

Partition a square matrix as $\mathbf{M} = \left(\begin{smallmatrix}\mathbf{A} & \mathbf{B}\\ \mathbf{C} & \mathbf{D}\end{smallmatrix}\right)$. When $\mathbf{D}$ is invertible, the *Schur complement* of $\mathbf{D}$ in $\mathbf{M}$ is $\mathbf{M}/\mathbf{D} = \mathbf{A} - \mathbf{B}\mathbf{D}^{-1}\mathbf{C}$, and when $\mathbf{A}$ is invertible, the Schur complement of $\mathbf{A}$ is $\mathbf{M}/\mathbf{A} = \mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B}$.

Lemma B.4 (Block factorization)**.** *If* $\mathbf{D}$ *is invertible,*

$$\mathbf{M} = \begin{pmatrix}\mathbf{I} & \mathbf{B}\mathbf{D}^{-1}\\ \mathbf{0} & \mathbf{I}\end{pmatrix}\begin{pmatrix}\mathbf{M}/\mathbf{D} & \mathbf{0}\\ \mathbf{0} & \mathbf{D}\end{pmatrix}\begin{pmatrix}\mathbf{I} & \mathbf{0}\\ \mathbf{D}^{-1}\mathbf{C} & \mathbf{I}\end{pmatrix},$$

so $\det\mathbf{M} = \det(\mathbf{M}/\mathbf{D})\,\det\mathbf{D}$*, and when* $\mathbf{M}/\mathbf{D}$ *is invertible,*

$$\mathbf{M}^{-1} = \begin{pmatrix}(\mathbf{M}/\mathbf{D})^{-1} & -(\mathbf{M}/\mathbf{D})^{-1}\mathbf{B}\mathbf{D}^{-1}\\ -\mathbf{D}^{-1}\mathbf{C}(\mathbf{M}/\mathbf{D})^{-1} & \mathbf{D}^{-1} + \mathbf{D}^{-1}\mathbf{C}(\mathbf{M}/\mathbf{D})^{-1}\mathbf{B}\mathbf{D}^{-1}\end{pmatrix}.$$

The same holds with the roles of the blocks exchanged: if $\mathbf{A}$ *is invertible,* $\det\mathbf{M} = \det\mathbf{A}\,\det(\mathbf{M}/\mathbf{A})$ *and the lower right block of* $\mathbf{M}^{-1}$ *is* $(\mathbf{M}/\mathbf{A})^{-1}$*.*

Proof. Multiplying the last two factors gives $\left(\begin{smallmatrix}\mathbf{M/D} & \mathbf{0}\\ \mathbf{C} & \mathbf{D}\end{smallmatrix}\right)$, and the first factor adds $\mathbf{BD}^{-1}$ times the second row to the first, restoring $\mathbf{A} = \mathbf{M/D} + \mathbf{BD}^{-1}\mathbf{C}$ and $\mathbf{B}$. The outer factors are triangular with unit diagonal, so their determinants are one and their inverses change the sign of the off-diagonal block. Inverting the product in reverse order gives $\mathbf{M}^{-1}$ as stated. The exchanged version is the same argument with the factorization pivoting on $\mathbf{A}$. □

Two identities follow from computing one block of $\mathbf{M}^{-1}$ in two ways. They are the algebra behind every rank-one update in the book.

Lemma B.5 (Woodbury and the determinant lemma)**.** *Let* $\mathbf{A}$ *$(n\times n)$ and* $\mathbf{W}$ *$(k\times k)$ be invertible, and* $\mathbf{U}$ *$(n\times k)$ and* $\mathbf{V}$ *$(k\times n)$ arbitrary. If* $\mathbf{W}^{-1}+\mathbf{VA}^{-1}\mathbf{U}$ *is invertible, then*

$$(\mathbf{A}+\mathbf{UWV})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{U}\,(\mathbf{W}^{-1}+\mathbf{VA}^{-1}\mathbf{U})^{-1}\,\mathbf{VA}^{-1}, \tag{B.3}$$

$$\det(\mathbf{A}+\mathbf{UWV}) = \det\mathbf{A}\,\det\mathbf{W}\,\det(\mathbf{W}^{-1}+\mathbf{VA}^{-1}\mathbf{U}). \tag{B.4}$$

For a rank-one change, $\mathbf{U}=\mathbf{h}$, $\mathbf{V}=\mathbf{h}^{\mathsf T}$ *and* $\mathbf{W}=w$, *these read*

$$(\mathbf{A}+w\,\mathbf{hh}^{\mathsf T})^{-1} = \mathbf{A}^{-1} - \frac{w\,\mathbf{A}^{-1}\mathbf{hh}^{\mathsf T}\mathbf{A}^{-1}}{1+w\,\mathbf{h}^{\mathsf T}\mathbf{A}^{-1}\mathbf{h}}, \qquad \det(\mathbf{A}+w\,\mathbf{hh}^{\mathsf T}) = \det\mathbf{A}\,(1+w\,\mathbf{h}^{\mathsf T}\mathbf{A}^{-1}\mathbf{h}),$$

the Sherman–Morrison formula and the matrix determinant lemma.

Proof. Apply Lemma B.4 to $\mathbf{M} = \left(\begin{smallmatrix}\mathbf{A} & \mathbf{U}\\ -\mathbf{V} & \mathbf{W}^{-1}\end{smallmatrix}\right)$. Pivoting on the lower right block, $\mathbf{M/W}^{-1} = \mathbf{A}+\mathbf{UWV}$, so the upper left block of $\mathbf{M}^{-1}$ is $(\mathbf{A}+\mathbf{UWV})^{-1}$ and $\det\mathbf{M} = \det(\mathbf{A}+\mathbf{UWV})\,\det\mathbf{W}^{-1}$. Pivoting on the upper left block, $\mathbf{M/A} = \mathbf{W}^{-1}+\mathbf{VA}^{-1}\mathbf{U}$, so $\det\mathbf{M} = \det\mathbf{A}\,\det(\mathbf{W}^{-1}+\mathbf{VA}^{-1}\mathbf{U})$, and the upper left block of $\mathbf{M}^{-1}$, computed from that factorization, is the right side of equation (B.3). Equating the two expressions for each quantity gives both identities. □

Lemma B.6 (Push-through)**.** *For* $\mathbf{\Lambda}$ *and* $\mathbf{R}$ *positive definite and any* $\mathbf{H}$,

$$(\mathbf{\Lambda}+\mathbf{H}^{\mathsf T}\mathbf{R}^{-1}\mathbf{H})^{-1}\mathbf{H}^{\mathsf T}\mathbf{R}^{-1} = \mathbf{\Lambda}^{-1}\mathbf{H}^{\mathsf T}(\mathbf{R}+\mathbf{H}\mathbf{\Lambda}^{-1}\mathbf{H}^{\mathsf T})^{-1}.$$

Proof. $(\mathbf{\Lambda}+\mathbf{H}^{\mathsf T}\mathbf{R}^{-1}\mathbf{H})\,\mathbf{\Lambda}^{-1}\mathbf{H}^{\mathsf T} = \mathbf{H}^{\mathsf T}+\mathbf{H}^{\mathsf T}\mathbf{R}^{-1}\mathbf{H}\mathbf{\Lambda}^{-1}\mathbf{H}^{\mathsf T} = \mathbf{H}^{\mathsf T}\mathbf{R}^{-1}(\mathbf{R}+\mathbf{H}\mathbf{\Lambda}^{-1}\mathbf{H}^{\mathsf T})$. Multiply on the left by the inverse of the first factor and on the right by the inverse of the last. □

B.3 Marginals and conditionals

Partition $\mathbf{z}=(\mathbf{x},\mathbf{u})$, and partition $\boldsymbol{\mu}$, $\mathbf{\Sigma}$, $\boldsymbol{\eta}$ and $\mathbf{\Lambda}$ to match. Chapter 6 proved the information-form results: the marginal of $\mathbf{u}$ has the Schur complement $\mathbf{\Lambda}_{uu}-\mathbf{\Lambda}_{ux}\mathbf{\Lambda}_{xx}^{-1}\mathbf{\Lambda}_{xu}$ as its precision (Proposition 6.3), and conditioning on $\mathbf{x}$ keeps the block $\mathbf{\Lambda}_{uu}$ and shifts the information vector (Proposition 6.5). The moment form is the mirror image.

Proposition B.1 (Marginal and conditional, moment form)**.** *Let* $\mathbf{z}=(\mathbf{x},\mathbf{u})\sim\mathcal{N}(\boldsymbol{\mu},\mathbf{\Sigma})$ *with* $\mathbf{\Sigma}_{xx}$ *positive definite. The marginal of* $\mathbf{u}$ *is* $\mathcal{N}(\boldsymbol{\mu}_u,\mathbf{\Sigma}_{uu})$, *and the conditional of* $\mathbf{u}$ *given* $\mathbf{x}$ *is*

$$\mathbf{u}\mid\mathbf{x} \;\sim\; \mathcal{N}\big(\boldsymbol{\mu}_u+\mathbf{\Sigma}_{ux}\mathbf{\Sigma}_{xx}^{-1}(\mathbf{x}-\boldsymbol{\mu}_x),\quad \mathbf{\Sigma}_{uu}-\mathbf{\Sigma}_{ux}\mathbf{\Sigma}_{xx}^{-1}\mathbf{\Sigma}_{xu}\big). \tag{B.5}$$

Proof. The marginal is Lemma B.2. For the conditional, let $\mathbf{e}=\mathbf{u}-\boldsymbol{\mu}_u-\mathbf{\Sigma}_{ux}\mathbf{\Sigma}_{xx}^{-1}(\mathbf{x}-\boldsymbol{\mu}_x)$. The pair $(\mathbf{e},\mathbf{x})$ is an affine map of $\mathbf{z}$, hence Gaussian, and $\operatorname{Cov}(\mathbf{e},\mathbf{x}) = \mathbf{\Sigma}_{ux}-\mathbf{\Sigma}_{ux}\mathbf{\Sigma}_{xx}^{-1}\mathbf{\Sigma}_{xx} = \mathbf{0}$, so by Lemma B.3 $\mathbf{e}$ is independent of $\mathbf{x}$. It has mean zero and covariance $\mathbf{\Sigma}_{uu}-2\mathbf{\Sigma}_{ux}\mathbf{\Sigma}_{xx}^{-1}\mathbf{\Sigma}_{xu}+\mathbf{\Sigma}_{ux}\mathbf{\Sigma}_{xx}^{-1}\mathbf{\Sigma}_{xx}\mathbf{\Sigma}_{xx}^{-1}\mathbf{\Sigma}_{xu} = \mathbf{\Sigma}_{uu}-\mathbf{\Sigma}_{ux}\mathbf{\Sigma}_{xx}^{-1}\mathbf{\Sigma}_{xu}$. Given $\mathbf{x}$, therefore, $\mathbf{u}$ is a fixed vector plus the independent Gaussian $\mathbf{e}$, which is equation (B.5). □

Table B.1: Marginalizing and conditioning in the two forms. What is a selection of blocks in one form is a Schur complement in the other.

	moment form $(\boldsymbol{\mu}, \boldsymbol{\Sigma})$	information form $(\boldsymbol{\eta}, \boldsymbol{\Lambda})$
marginal of $\mathbf{u}$	$\boldsymbol{\mu}_u,\ \boldsymbol{\Sigma}_{uu}$	$\boldsymbol{\eta}_u - \boldsymbol{\Lambda}_{ux}\boldsymbol{\Lambda}_{xx}^{-1}\boldsymbol{\eta}_x,\ \boldsymbol{\Lambda}_{uu} - \boldsymbol{\Lambda}_{ux}\boldsymbol{\Lambda}_{xx}^{-1}\boldsymbol{\Lambda}_{xu}$
conditional on $\mathbf{x}$	$\boldsymbol{\mu}_u + \boldsymbol{\Sigma}_{ux}\boldsymbol{\Sigma}_{xx}^{-1}(\mathbf{x} - \boldsymbol{\mu}_x),\ \boldsymbol{\Sigma}_{uu} - \boldsymbol{\Sigma}_{ux}\boldsymbol{\Sigma}_{xx}^{-1}\boldsymbol{\Sigma}_{xu}$	$\boldsymbol{\eta}_u - \boldsymbol{\Lambda}_{ux}\mathbf{x},\ \boldsymbol{\Lambda}_{uu}$

The proof never inverts $\boldsymbol{\Sigma}$ or $\boldsymbol{\Sigma}_{uu}$, so the proposition holds for any covariance whose conditioning block $\boldsymbol{\Sigma}_{xx}$ is positive definite. When $\boldsymbol{\Sigma}$ itself is positive definite the two forms agree through Lemma B.4. The upper left block of $\boldsymbol{\Sigma}^{-1}$, ordered as $(\mathbf{u}, \mathbf{x})$, is $\boldsymbol{\Lambda}_{uu} = (\boldsymbol{\Sigma}_{uu} - \boldsymbol{\Sigma}_{ux}\boldsymbol{\Sigma}_{xx}^{-1}\boldsymbol{\Sigma}_{xu})^{-1}$, the inverse of the conditional covariance, as Proposition 6.5 requires. Symmetrically, $\boldsymbol{\Sigma}_{uu}$ is the inverse of the Schur complement of $\boldsymbol{\Lambda}_{xx}$, as Proposition 6.3 requires. Table B.1 sets the four results side by side.

B.4 Products and linear-Gaussian models

Proposition B.2 (Products of Gaussians)**.** *As functions of* $\mathbf{z}$, $\prod_i \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ *is proportional to the Gaussian with precision* $\boldsymbol{\Lambda} = \sum_i \boldsymbol{\Sigma}_i^{-1}$ *and information vector* $\boldsymbol{\eta} = \sum_i \boldsymbol{\Sigma}_i^{-1}\boldsymbol{\mu}_i$*. For two factors the constant of proportionality is*

$$\int \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)\, \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)\, \mathrm{d}\mathbf{z} \;=\; \mathcal{N}(\boldsymbol{\mu}_1;\ \boldsymbol{\mu}_2,\ \boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_2).$$

Proof. In information form the exponents add, and the corollary of Lemma B.1 reads off the product. For the constant, let $\mathbf{z} \sim \mathcal{N}(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$ and $\mathbf{y} = \mathbf{z} + \boldsymbol{\nu}$ with an independent $\boldsymbol{\nu} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_1)$. The density of $\mathbf{y}$ at $\boldsymbol{\mu}_1$ is $\int \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)\, \mathcal{N}(\boldsymbol{\mu}_1; \mathbf{z}, \boldsymbol{\Sigma}_1)\, \mathrm{d}\mathbf{z}$, which is the integral above, since $\mathcal{N}(\boldsymbol{\mu}_1; \mathbf{z}, \boldsymbol{\Sigma}_1) = \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$. By Lemma B.2 $\mathbf{y} \sim \mathcal{N}(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_2)$. □

The product rule is the one sentence Chapter 6 builds on: every factor adds its precision and its information vector.

Proposition B.3 (The linear-Gaussian model)**.** *Let* $\mathbf{z} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$*, with* $\boldsymbol{\Sigma}$ *positive semidefinite, and let* $\mathbf{y} = \mathbf{H}\mathbf{z} + \mathbf{c} + \boldsymbol{\nu}$ *with* $\boldsymbol{\nu} \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$ *independent of* $\mathbf{z}$ *and* $\mathbf{R}$ *positive definite. Write* $\mathbf{S} = \mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^{\mathsf{T}} + \mathbf{R}$ *and* $\mathbf{K} = \boldsymbol{\Sigma}\mathbf{H}^{\mathsf{T}}\mathbf{S}^{-1}$*, the covariance of the innovation* $\mathbf{y} - \mathbf{H}\boldsymbol{\mu} - \mathbf{c}$ *and the gain. (This* $\mathbf{S}$ *is not the map* $\mathbf{z} = \mathbf{S}\mathbf{u}$ *of §B.1.) Then:*

1. *the pair* $(\mathbf{z}, \mathbf{y})$ *is Gaussian with cross-covariance* $\mathrm{Cov}(\mathbf{z}, \mathbf{y}) = \boldsymbol{\Sigma}\mathbf{H}^{\mathsf{T}}$*;*
2. *the* predictive *distribution of the reading is* $\mathbf{y} \sim \mathcal{N}(\mathbf{H}\boldsymbol{\mu} + \mathbf{c},\ \mathbf{S})$*;*
3. *the posterior is*
$$\mathbf{z} \mid \mathbf{y} \;\sim\; \mathcal{N}(\boldsymbol{\mu} + \mathbf{K}(\mathbf{y} - \mathbf{H}\boldsymbol{\mu} - \mathbf{c}),\quad \boldsymbol{\Sigma} - \mathbf{K}\mathbf{H}\boldsymbol{\Sigma}); \tag{B.6}$$
4. *if* $\boldsymbol{\Sigma}$ *is positive definite, the posterior has precision* $\boldsymbol{\Lambda} + \mathbf{H}^{\mathsf{T}}\mathbf{R}^{-1}\mathbf{H}$ *and information vector* $\boldsymbol{\eta} + \mathbf{H}^{\mathsf{T}}\mathbf{R}^{-1}(\mathbf{y} - \mathbf{c})$*, with* $\boldsymbol{\Lambda} = \boldsymbol{\Sigma}^{-1}$ *and* $\boldsymbol{\eta} = \boldsymbol{\Lambda}\boldsymbol{\mu}$*.*

Proof. The pair $(\mathbf{z}, \boldsymbol{\nu})$ is Gaussian with block diagonal covariance, and $(\mathbf{z}, \mathbf{y})$ is an affine map of it, so items 1 and 2 follow from Lemma B.2; the cross-covariance is $\mathbb{E}[(\mathbf{z} - \boldsymbol{\mu})(\mathbf{H}(\mathbf{z} - \boldsymbol{\mu}) + \boldsymbol{\nu})^{\mathsf{T}}] = \boldsymbol{\Sigma}\mathbf{H}^{\mathsf{T}}$. Item 3 is Proposition B.1 with $\mathbf{x} = \mathbf{y}$ and $\mathbf{u} = \mathbf{z}$, whose conditioning block is $\mathbf{S}$, positive definite because $\mathbf{R}$ is. For item 4, the posterior precision claimed inverts to the posterior covariance of item 3 by equation (B.3) with $\mathbf{A} = \boldsymbol{\Lambda}$, $\mathbf{U} = \mathbf{H}^{\mathsf{T}}$, $\mathbf{W} = \mathbf{R}^{-1}$ and $\mathbf{V} = \mathbf{H}$:

$(\mathbf{\Lambda} + \mathbf{H}^\mathsf{T}\mathbf{R}^{-1}\mathbf{H})^{-1} = \mathbf{\Sigma} - \mathbf{\Sigma}\mathbf{H}^\mathsf{T}\mathbf{S}^{-1}\mathbf{H}\mathbf{\Sigma} = \mathbf{\Sigma} - \mathbf{K}\mathbf{H}\mathbf{\Sigma}$. The mean is that covariance times the claimed information vector. Its first part is $(\mathbf{\Sigma} - \mathbf{K}\mathbf{H}\mathbf{\Sigma})\mathbf{\Lambda}\boldsymbol{\mu} = \boldsymbol{\mu} - \mathbf{K}\mathbf{H}\boldsymbol{\mu}$, and by Lemma B.6 its second part is $(\mathbf{\Lambda} + \mathbf{H}^\mathsf{T}\mathbf{R}^{-1}\mathbf{H})^{-1}\mathbf{H}^\mathsf{T}\mathbf{R}^{-1}(\mathbf{y} - \mathbf{c}) = \mathbf{K}(\mathbf{y} - \mathbf{c})$. Their sum is the mean of item 3. □

With one reading, $\mathbf{H} = \mathbf{h}^\mathsf{T}$ and $\mathbf{R} = \sigma^2$, item 3 is Proposition 3.1 and Proposition 11.1, and item 4 is Proposition 3.2. The proof here conditions the joint distribution rather than completing a square, so it never inverts $\mathbf{\Sigma}$: the one-reading formulas hold for a singular covariance such as the rank-two $\mathbf{\Sigma}_z$ of §3.4, exactly and not only as a limit.

B.5 Rank-one changes

A reading of $\mathbf{h}^\mathsf{T}\mathbf{z}$ with weight $w = 1/\sigma^2$ changes the precision by the rank-one term $w\,\mathbf{h}\mathbf{h}^\mathsf{T}$. With $\mathbf{s} = \mathbf{\Sigma}\mathbf{h}$, Lemma B.5 gives the new covariance and determinant at the cost of one solve:

$$\mathbf{\Sigma}' = \mathbf{\Sigma} - \frac{w\,\mathbf{s}\mathbf{s}^\mathsf{T}}{1 + w\,\mathbf{h}^\mathsf{T}\mathbf{s}}, \qquad \det \mathbf{\Lambda}' = \det \mathbf{\Lambda}\,(1 + w\,\mathbf{h}^\mathsf{T}\mathbf{s}). \tag{B.7}$$

A reading can also be removed, which is how a leave-one-out residual or a rejected meter is computed without refactoring. Removing it subtracts the same term, $\mathbf{\Lambda} = \mathbf{\Lambda}' - w\,\mathbf{h}\mathbf{h}^\mathsf{T}$, and with $\mathbf{s}' = \mathbf{\Sigma}'\mathbf{h}$ equation (B.3) with $\mathbf{W} = -w$ gives

$$\mathbf{\Sigma} = \mathbf{\Sigma}' + \frac{w\,\mathbf{s}'\mathbf{s}'^\mathsf{T}}{1 - w\,\mathbf{h}^\mathsf{T}\mathbf{s}'}, \tag{B.8}$$

valid exactly when $1 - w\,\mathbf{h}^\mathsf{T}\mathbf{s}' > 0$, which is the condition that the reduced precision stays positive definite. The determinant identity gives what a reading is worth before it is taken, measured in information rather than variance. With the entropy of equation (B.11) below, the entropy of $\mathbf{z}$ falls by

$$\tfrac{1}{2}\log\det\mathbf{\Sigma} - \tfrac{1}{2}\log\det\mathbf{\Sigma}' \;=\; \tfrac{1}{2}\log(1 + \mathbf{h}^\mathsf{T}\mathbf{\Sigma}\mathbf{h}/\sigma^2), \tag{B.9}$$

the mutual information between $\mathbf{z}$ and the reading. It depends on the reading only through the ratio of the variance it predicts, $\mathbf{h}^\mathsf{T}\mathbf{\Sigma}\mathbf{h}$, to its noise, and it is the information counterpart of the variance reduction of equation (11.3).

B.6 Integrals, quadratic forms and divergences

Proposition B.4 (The evidence of a quadratic objective)**.** *If* $C(\mathbf{z}) = C^\star + \frac{1}{2}(\mathbf{z} - \mathbf{z}^\star)^\mathsf{T}\mathbf{\Lambda}(\mathbf{z} - \mathbf{z}^\star)$ *with* $\mathbf{\Lambda}$ *positive definite, then* $\int \exp(-C(\mathbf{z}))\,\mathrm{d}\mathbf{z} = \exp(-C^\star)\,(2\pi)^{n/2}\,(\det\mathbf{\Lambda})^{-1/2}$.

Proof. Lemma B.1, with the square already completed. □

For a linear-Gaussian model the negative log posterior of equation (6.4) is exactly such a quadratic. The joint density of the unknowns and the readings is $\exp(-C)$ times the normalizing constants of the factors, so the evidence is $\exp(-C(\mathbf{z}^\star))$, times $(2\pi)^{n/2}(\det\mathbf{\Lambda})^{-1/2}$, times those constants. Its logarithm is equation (7.2) of Chapter 7, whose constant collects $\frac{n}{2}\log 2\pi$ and the logarithms of the factors' normalizers.

Proposition B.5 (Quadratic forms and the chi-squared law)**.** *Let* $\mathbf{z}$ *have mean* $\boldsymbol{\mu}$ *and covariance* $\boldsymbol{\Sigma}$*. For any symmetric* $\mathbf{A}$ *and any* $\mathbf{m}$*,*

$$\mathbb{E}\big[(\mathbf{z}-\mathbf{m})^{\mathsf{T}}\mathbf{A}(\mathbf{z}-\mathbf{m})\big] = \operatorname{tr}(\mathbf{A}\boldsymbol{\Sigma}) + (\boldsymbol{\mu}-\mathbf{m})^{\mathsf{T}}\mathbf{A}(\boldsymbol{\mu}-\mathbf{m}). \tag{B.10}$$

If $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_m)$ *and* $\mathbf{P}$ *is a symmetric idempotent matrix of rank* r*, then* $\boldsymbol{\epsilon}^{\mathsf{T}}\mathbf{P}\boldsymbol{\epsilon}$ *is chi-squared with* r *degrees of freedom, with mean* r *and variance* $2r$*.*

Proof. Write $\mathbf{z}-\mathbf{m} = (\mathbf{z}-\boldsymbol{\mu})+(\boldsymbol{\mu}-\mathbf{m})$. The cross term has zero mean, and $\mathbb{E}[(\mathbf{z}-\boldsymbol{\mu})^{\mathsf{T}}\mathbf{A}(\mathbf{z}-\boldsymbol{\mu})] = \operatorname{tr}(\mathbf{A}\,\mathbb{E}[(\mathbf{z}-\boldsymbol{\mu})(\mathbf{z}-\boldsymbol{\mu})^{\mathsf{T}}]) = \operatorname{tr}(\mathbf{A}\boldsymbol{\Sigma})$, since a scalar equals its trace and the trace is invariant under cyclic permutation. For the second part, a symmetric idempotent matrix has eigenvalues zero and one, so $\mathbf{P} = \mathbf{Q}\mathbf{D}\mathbf{Q}^{\mathsf{T}}$ with $\mathbf{Q}$ orthogonal and $\mathbf{D}$ diagonal with r ones. By Lemma B.2, $\boldsymbol{\delta} = \mathbf{Q}^{\mathsf{T}}\boldsymbol{\epsilon}$ is again standard Gaussian, and $\boldsymbol{\epsilon}^{\mathsf{T}}\mathbf{P}\boldsymbol{\epsilon} = \sum_{i\le r}\delta_i^2$, a sum of r independent squared standard normals. Each has mean one and, since $\mathbb{E}[\delta_i^4] = 3$, variance two. □

The second part is the last step of Proposition 4.2 and of Proposition 12.1. There the matrix is the projector $\mathbf{I}-\mathbf{P}$ onto the complement of the column space of the standardized Jacobian, of rank $m-n$.

The *entropy* of a density is $H[p] = -\mathbb{E}_p[\log p]$, and the *Kullback–Leibler divergence* of q from p is $\mathrm{KL}(q\,\|\,p) = \mathbb{E}_q[\log q - \log p]$, which is non-negative and zero only when $q = p$.

Proposition B.6 (Entropy and divergence of Gaussians)**.** *For* $\boldsymbol{\Sigma}$ *positive definite,*

$$H\big[\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})\big] = \tfrac{1}{2}\log\det(2\pi e\,\boldsymbol{\Sigma}), \tag{B.11}$$

and for two Gaussians $\mathcal{N}_0 = \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$ *and* $\mathcal{N}_1 = \mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ *on* $\mathbb{R}^n$*,*

$$\mathrm{KL}(\mathcal{N}_0\,\|\,\mathcal{N}_1) = \tfrac{1}{2}\Big[\operatorname{tr}(\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\Sigma}_0) + (\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)^{\mathsf{T}}\boldsymbol{\Sigma}_1^{-1}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_0) - n + \log\det\boldsymbol{\Sigma}_1 - \log\det\boldsymbol{\Sigma}_0\Big]. \tag{B.12}$$

Proof. From equation (B.1), $-\log\mathcal{N}_1(\mathbf{z}) = \frac{n}{2}\log 2\pi + \frac{1}{2}\log\det\boldsymbol{\Sigma}_1 + \frac{1}{2}(\mathbf{z}-\boldsymbol{\mu}_1)^{\mathsf{T}}\boldsymbol{\Sigma}_1^{-1}(\mathbf{z}-\boldsymbol{\mu}_1)$. Its expectation under $\mathcal{N}_0$ is, by equation (B.10), $\frac{n}{2}\log 2\pi + \frac{1}{2}\log\det\boldsymbol{\Sigma}_1 + \frac{1}{2}\operatorname{tr}(\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\Sigma}_0) + \frac{1}{2}(\boldsymbol{\mu}_0-\boldsymbol{\mu}_1)^{\mathsf{T}}\boldsymbol{\Sigma}_1^{-1}(\boldsymbol{\mu}_0-\boldsymbol{\mu}_1)$. With $\mathcal{N}_1 = \mathcal{N}_0$ the trace is n, which gives the entropy, $\frac{n}{2}\log 2\pi + \frac{1}{2}\log\det\boldsymbol{\Sigma}_0 + \frac{n}{2}$. The divergence is the second expectation minus the entropy of $\mathcal{N}_0$. □

Proposition 9.2 minimizes equation (B.12) with $\mathcal{N}_1$ the posterior, over $\mathcal{N}_0$ a product of independent Gaussians. The terms that depend on $\mathcal{N}_0$ are $\operatorname{tr}(\boldsymbol{\Lambda}\boldsymbol{\Sigma}_0) + (\boldsymbol{\mu}-\boldsymbol{\mu}_0)^{\mathsf{T}}\boldsymbol{\Lambda}(\boldsymbol{\mu}-\boldsymbol{\mu}_0) - \log\det\boldsymbol{\Sigma}_0$, and with a diagonal $\boldsymbol{\Sigma}_0$ the trace keeps only the diagonal of $\boldsymbol{\Lambda}$.

B.7 The Laplace approximation

Let a posterior be $p(\mathbf{z}\mid\mathbf{y}) \propto \exp(-C(\mathbf{z}))$ with C twice differentiable, a minimum at $\mathbf{z}^\star$, and a positive definite Hessian $\nabla^2 C^\star = \nabla^2 C(\mathbf{z}^\star)$ there. Because the gradient vanishes at the minimum, Taylor's theorem gives $C(\mathbf{z}) = C(\mathbf{z}^\star) + \frac{1}{2}(\mathbf{z}-\mathbf{z}^\star)^{\mathsf{T}}\,\nabla^2 C^\star\,(\mathbf{z}-\mathbf{z}^\star) + O(\|\mathbf{z}-\mathbf{z}^\star\|^3)$. Dropping the cubic remainder and applying the corollary of Lemma B.1 and Proposition B.4 gives the *Laplace approximation* to the posterior and to the evidence,

$$p(\mathbf{z}\mid\mathbf{y}) \approx \mathcal{N}\big(\mathbf{z}^\star,\ (\nabla^2 C^\star)^{-1}\big), \qquad \log\int \exp(-C)\,\mathrm{d}\mathbf{z} \approx -C(\mathbf{z}^\star) + \tfrac{n}{2}\log 2\pi - \tfrac{1}{2}\log\det\nabla^2 C^\star. \tag{B.13}$$

Table B.2: The identities of this appendix and where the book uses them.

identity	used in
Gaussian integral, reading a Gaussian off its exponent (Lemma B.1)	every posterior of Chapters 3 to 9; evidence, equation (7.2)
affine maps, singular covariances (Definition B.1, Lemma B.2)	$\mathbf{z} = \mathbf{S}\mathbf{u}$ in §3.4; sampling, Proposition 6.4; misfit law
block factorization, Schur complement (Lemma B.4)	marginalization and region messages, Chapters 6 and 8
Woodbury, Sherman–Morrison, determinant lemma (Lemma B.5)	Propositions 3.1 and 11.1; sequential updates, Chapter 11
moment-form conditional (Proposition B.1)	conditioning a joint prediction on a reading; virtual sensors
product rule (Proposition B.2)	information form, Proposition 3.2; Exercise 6.1
linear-Gaussian model (Proposition B.3)	predictive checks and innovations, Chapters 11 and 12; one reading with a singular prior
rank-one update, downdate, information gain (§B.5)	the value of a reading, equation (11.3); removing a meter
quadratic forms, chi-squared (Proposition B.5)	the misfit law, Propositions 4.2 and 12.1
entropy and divergence (Proposition B.6)	mean field, Proposition 9.2
Laplace approximation, Gauss–Newton Hessian (§B.7)	equation (6.9); branch evidence, Chapter 7

Both are exact when C is quadratic. Otherwise their accuracy depends on how much of the posterior's mass lies where the cubic remainder is not small against the quadratic, which shrinks as the posterior concentrates [91].

The book's objective has the least-squares form $C = \frac{1}{2}\mathbf{g}^{\mathsf{T}}\boldsymbol{\Omega}\mathbf{g}$ of equation (6.4), with $\boldsymbol{\Omega} = \operatorname{diag}(w_k)$. Differentiating twice,

$$\nabla C = \mathbf{J}_g^{\mathsf{T}}\boldsymbol{\Omega}\mathbf{g}, \qquad \nabla^2 C = \mathbf{J}_g^{\mathsf{T}}\boldsymbol{\Omega}\mathbf{J}_g + \sum_k w_k\, g_k\, \nabla^2 g_k. \tag{B.14}$$

The first term is the Gauss–Newton matrix $\boldsymbol{\Lambda} = \mathbf{J}_g^{\mathsf{T}}\boldsymbol{\Omega}\mathbf{J}_g$ that the book's solver assembles and that equation (6.9) uses; the second is dropped. For row k the dropped term is smaller than the kept one by roughly the factor $|g_k|\,\|\nabla^2 g_k\|/\|\nabla g_k\|^2$: the row's residual at the mode times its curvature, over its squared slope. It vanishes for linear rows. It is small for rows that are nearly satisfied at the mode, such as physics rows held to a small width, and for readings that the posterior fits to within their noise, and §6.2.1 discusses when that fails. The Gauss–Newton matrix is also positive semidefinite by construction, which the full Hessian need not be away from the mode.

B.8 Where the book uses each identity

Table B.2 lists, for each result of this appendix, the places in the book that rely on it.

Appendix C

From the book to the engine

The constructions of this book are implemented in TERRA, a domain-independent engine for multi-conservation graphs, written by the author. Nothing in Chapters 1 to 18 depends on it: their scripts import no module of the engine and run on a numerical library alone. The case studies of Part VI were run with it. This appendix says what the engine is, shows a model written in it, maps each construction of the book to the module that implements it, and says what the case studies used and how to rerun them.

C.1 What TERRA is

TERRA is the Python package `terra_graph_engine`. It needs Python 3.11 or later, NumPy, SciPy and Jinja2. It has two layers.

The *engine* knows nothing of any domain. It holds the objects of Part I: nodes with roles, edges with one channel per conserved quantity, closures, boundary conditions, and the compiled residual system with its sparse Jacobian. It solves that system by Newton's method in steady state and by a BDF integrator in time. On the same rows it builds the probabilistic model of Parts II and III, together with counterfactual branching and hierarchical decomposition.

The *plugins* carry what is specific to one domain or one method. There are foundation plugins for substance physics and electrical quantities, and domain plugins for hydraulics, HVAC, power flow, transmission grids, batteries and data centers. There are also method plugins for reliability enumeration and for the census and its surrogate. A plugin supplies closures, factories that build standard networks, and algorithms that sit on top of the engine. The studies of Part VI use the `hydraulics` plugin, which carries the head-loss closures, the pump and valve models, the readers for EPANET input files and for the published leak-detection benchmarks, the leak-hypothesis ranking and screening, and the properties of water, over the `thermo` foundation plugin that supplies those properties. Some plugin algorithms in other domains, such as the DC dispatch linear programs of the power-flow plugin, solve their own problems directly and never call the engine's Newton solver; no algorithm used in Part VI does.

The objects a reader meets first are these.

- `Model` is the builder. Nodes, edges, closures and initial values are added to it, and `build()` compiles it.
- `CompiledModel` is the frozen result: a registry of unknowns, the residual rows of Chapter 2, and their Jacobian.

- `tge.solve` solves a compiled model in steady state, by Newton's method with a backtracking line search. `tge.simulate` integrates it in time.
- `InferenceModel` views a compiled model as the Gaussian factor graph of Chapter 3. Its physics rows are weighted by a physics width, `physics_sigma`, which is the soft-constraint width of Chapter 4. `observe` attaches a sensor, `prior` a belief, and `add_latent_fault` an additive unknown source on a balance.
- `infer` returns a `Posterior`: the MAP state by Gauss–Newton weighted least squares, and the Laplace covariance from the information matrix, as in Chapter 6.

A prior is either a datum or a regularizer. A datum is a belief whose error is a draw at its stated spread, such as a forecast. A regularizer is a broad prior that only keeps an unknown identifiable. Both act the same way in the solve. The adequacy test of Chapter 12 counts only data, for the reason §12.3 gives.

C.2 A model in a dozen lines

The smallest model with physics and redundancy has one closure, $y = 2x$, three readings of x, and nothing observed about y except a broad regularizing prior.

```
import terra_graph_engine as tge
from terra_graph_engine import Model, NodeRole, PropertyClosure
from terra_graph_engine.inference import InferenceModel, infer, goodness_of_fit

m = Model()
m.add_node("n", role=NodeRole.RESERVOIR)
m.add_closure(PropertyClosure(out="y", inputs=["x"], f=lambda x: 2.0 * x,
                              analytic_jacobian=lambda x: {"x": 2.0}))
m.set_initial("x", 10.0)
m.set_initial("y", 20.0)
cm = m.build()

im = InferenceModel(cm, physics_sigma=1e-3)
for reading in (11.0, 9.0, 10.5):
    im.observe("x", reading, noise=1.0)
im.prior("y", mean=20.0, std=50.0, regularizer=True)
post = infer(im)
fit = goodness_of_fit(im, post)
```

The posterior gives $x = 10.167 \pm 0.577$, which is the mean of the three readings with spread $1/\sqrt{3}$. It gives $y = 20.333 \pm 1.154$, carried through the closure with twice the spread. The closure's row has put y on the same footing as x with no reading of y; this is the virtual sensor of Chapter 12. The misfit is 2.167 on 2 degrees of freedom, a p-value of 0.338. The count is one physics row plus three readings less two unknowns. The regularizing prior on y counts in neither the misfit nor the degrees of freedom.

C.3 Where each construction lives

Tables C.1 and C.2 map the chapters of Parts I to V to the modules that implement them. Engine modules are named relative to the package, and plugin modules by plugin. Where a chapter's construction has no module, its script carries it out directly.

Table C.1: Parts I to III: the constructions and the modules that implement them.

ch.	construction	modules
1	nodes and roles, edges and channels, boundary conditions, sources, losses, storage; conserved domains; closure patterns	`model/` (`node`, `edge`, `boundary`, `model`); `domains/`; `closure/` (carrier flow, gradient flow, property, piecewise)
2	incidence, the registry of unknowns, residual rows, the Jacobian, the Newton solve	`topology/graph`; `variables/registry`; `model/compiled_model`; `assembly/residual`, `assembly/jacobian`; `numerics/solver/newton`; `entry` (`solve`)
3	sensors, priors, latent faults on a balance	`inference/model` (`InferenceModel`); `inference/factors`; `model/compiled_model` (`with_latent_fault`)
4	physics as soft rows of a set width; bounds on a posterior	`inference/model` (`physics_sigma`); `inference/map_solve`; `inference/constrained`
5	separators and elimination on the information matrix	`inference/gaussian_bp`; `hierarchy/coordinator`
6	MAP by weighted least squares; the Laplace posterior	`inference/map_solve`; `inference/result` (`Posterior`)
7	hybrid states and branch mixtures; checks and corrections of the Laplace posterior	`closure/patterns/piecewise`, `numerics/regimes`; `inference/validity`, `inference/laplace_is`, `inference/particle`, `inference/escalation`; plugin `uncertainty` (`mixture`)
8	Schur complements over an interface; elimination orders	`inference/gaussian_bp`; `hierarchy/coordinator`
9	sampling corrections; iteration between regions	`inference/particle`, `inference/laplace_is`; `hierarchy/coupling`, `hierarchy/ports`
10	forward sampling through the physics; batched and swept solves	`inference/propagate`; `assembly/batch` (`solve_batch`); `inference/map_batch`; `numerics/sweep`
11	rank-one conditioning on new readings; particle filtering; the value of a reading	`inference/incremental`; `inference/particle`; `inference/online_identify`; `inference/experiment_design`

C.4 The case studies

Table C.3 lists, for each chapter of Part VI, what the study used and what one engine solve is in it. Every study in this part runs the engine's own Newton solver or its MAP inference on a compiled hydraulic model, so the solve counts the chapters report are directly comparable with one another. The two heating studies compile two conserved quantities onto the same nodes and edges and solve them as one residual system, not as two coupled ones.

Each study lives in its own directory under `docs/terra_graph_engine/case_studies/` in the repository that accompanies the book, in the group `water_networks` or `district_heating`. It is run from the repository root with the package source on the Python path,

Table C.2: Parts IV and V: the constructions and the modules that implement them.

ch.	construction	modules
12	the misfit law, studentized residuals, fault hypotheses by evidence, identifiability, virtual sensors	`inference/goodness_of_fit`; `inference/model_comparison`; `inference/identifiability`; `inference/queries` (`locate_faults`)
13	interventions on the model; stacked branches and their aggregation	`counterfactual/` (`perturbation`, `branch`, `reduce`)
14	sensitivities at a solution; the census of values and gradients, supporting planes, the certificate, the questions answered from it	`numerics/sensitivity`; `assembly/input_jacobian`; plugin `uncertainty` (`evaluator`, `pieces`, `queries`, `attribution`, `control_variate`, `sampling`)
15	component reliability, common cause, probability-ordered enumeration with certified bounds, interdiction	plugin `reliability` (`component_reliability`, `common_cause`, `enumeration`, `interdiction`, `risk`, `screening`, `rare_event`)
16	time integration and events; linearization, observers and model-predictive control	`entry` (`simulate`); `numerics/integrator/`; `assembly/dynamic`; `numerics/linearize`, `numerics/observer`, `numerics/mpc`; `controls/`; `inference/forecast`
17	subsystems with ports, interface coupling, Schur-complement coordination, composition	`hierarchy/` (`subsystem`, `ports`, `coupling`, `coordinator`, `compose`)
18	region messages, separator values and compression, region trees, certified boxing, boundary equivalents	plugin `reliability` (`local_global`, `compressibility`, `region_tree`, `boxing`, `environment_k`, `regional_contingency`, `thermal_adapter`); plugin `power_flow` (`regional_dc`, `partitioning`)

Table C.3: The case studies of Part VI: the engine and plugin pieces each used, and what one engine solve is.

ch.	engine and plugins	one engine solve
20	plugin `hydraulics` (`build_model`, `apply_flat_start`, the EPANET oracle `epanet_steady`); engine: `solve`, `infer`, `estimate`, `goodness_of_fit`	a Newton or MAP solve of the five-unknown model
21	plugin `hydraulics` (`leak_variable`, `leak_hypotheses`, `rank_leak_hypotheses`, `articulation_decomposition`); engine: `assemble_residual`, `assemble_jacobian`, `infer`	a MAP solve of the 211-unknown model with its latent demands
22	plugin `hydraulics` (`irofti` benchmark reader, `screen_leak_candidates`, `rank_leak_hypotheses`); engine: `infer`, `rank_expected_information_gain_measurements`	a MAP solve of the joint model over a window of snapshots; the screen costs one Gauss–Newton step against a factorization already held
23	plugin `hydraulics` (`battledim` readers and scorer, `calibration`, `build_joint_model`); engine: `InferenceModel`, `infer`	a MAP solve of the joint model over all of a scenario's snapshots, 59,325 unknowns at the largest

```
PYTHONPATH=src python \
  docs/terra_graph_engine/case_studies/<group>/<study>/run.py
```

and it writes a results file next to its script. The book's script for the chapter reads that file and runs no physics:

```
cd docs/books/hydraulic
python scripts/w3_battledim.py
```

Each chapter's last section gives its exact commands, its run time, and the tests that pin its committed results. The engine and its plugins carry their own test suite, under `tests/terra_graph_engine` and `tests/plugins`.

Appendix D

A catalogue of closures

Table 1.2 named four closure patterns. This appendix fills them in. It gathers the closures of the domains this book touches, thermal, fluid, moist air, electrical, storage and plant equipment, and gives each one in the same short form. Some are implemented in the plugins of TERRA, and the entry names the class or function; the forms given are the ones the code implements. Others are standard relations the plugins do not yet carry, and the entry cites a textbook. Either way the entry says what a modeler needs before writing the relation into a graph.

D.1 How to read an entry

Each entry gives the relation in residual form, zero when the relation holds, and then eight short fields.

- *Pattern*: carrier, gradient, property or piecewise, as in Table 1.2, or a custom closure when none fits.
- *Unknowns*: the variables the closure reads. Any of them can be an unknown; which ones are is a property of the question (Principle 1.2).
- *Parameters*: constants fixed when the closure is built, with units.
- *Jacobian*: the partial derivatives of the residual, analytic or by finite differences.
- *Kinks*: where the relation, or its derivative, is not smooth. A kink is where Newton's method, the Laplace approximation and the certificates of Chapter 14 need care (§14.5).
- *Shape*: whether the relation is monotone, linear, convex or none of these, in the variables that matter. Convexity is what the supporting planes of Proposition 14.2 need, and monotonicity is what the floors of Proposition 14.4 need.
- *Prior*: the parameter a study usually leaves uncertain, and why. This is a modeling judgment, not a property of the code.
- *Source*: the plugin and class that implements it, or the reference for a textbook relation.

Three kinds of physics look like closures and are not. A *storage relation* is the accumulation term of a balance, $\mathrm{d}X/\mathrm{d}t$, and belongs to the node (Chapter 16). A *pin* fixes a variable at a setpoint and is a one-variable row. And an *outer loop* changes the model between solves, as a generator that hits its reactive limit does. The entries say when a relation is one of these.

Table D.1 indexes the entries.

Table D.1: The closures of this appendix: pattern, whether the relation kinks, its shape, and where it comes from (P: a plugin of TERRA; T: a textbook relation).

closure	pattern	kinks	shape	src
Thermal				
conductance, $Q = UA\,\Delta T$	gradient	no	linear	P
series conduction	gradient	no	linear	P
convection with a correlation	gradient	yes	monotone in ΔT	T
radiation between two surfaces	custom	no	monotone, not convex	P
counterflow exchanger, fixed rates	custom	no	linear in T	P
counterflow exchanger, variable rates	custom	yes	not evident	P
cooling tower	gradient	yes	non-decreasing	P
waterside economizer	property	yes	non-decreasing	P
enthalpy carried by mass; throttle	carrier; custom	no	bilinear; linear	P
temperature-dependent load	property	no	linear	P
Fluid				
quadratic resistance; damper	gradient	yes	increasing, odd	P
water main (Hazen–Williams)	gradient	at zero flow	increasing, $m^{1.852}$	T
pipe friction (Darcy–Weisbach)	gradient	yes	increasing	T
laminar pipe (Hagen–Poiseuille)	gradient	no	linear	T
fan or pump curve	gradient	yes	decreasing in flow	P
fan and pump power	property	no	convex (cubic)	P
valve with a loss coefficient	gradient	yes	increasing in flow	T
check valve	piecewise	yes	monotone	T
isothermal gas pipe	gradient	yes	increasing	T
ideal-gas mass flow	property	no	linear	P
mixing of two streams	property	no	bilinear	P
Moist air				
humidity ratio and moist-air enthalpy	property	no	monotone	P
coil condensation	property	yes	non-decreasing	P
moisture carried by mass	carrier	no	bilinear	P
Electrical				
DC branch	gradient	no	linear	P
AC branch in polar form	custom	no	trigonometric	P
transformer with a tap	custom	no	trigonometric	P
slack and voltage-controlled buses	pin	—	—	P
reactive limits of a generator	outer loop	yes	—	P
transformer loss	property	no	convex	P
UPS loss	property	no	convex	P
PDU loss	property	no	linear	P
fixed share of a flow	custom	no	linear	P
generator fuel curve	property	gated	affine	P
Storage and electrochemistry				
battery energy with efficiencies	storage	no	linear	P
signed power port	property	yes	piecewise linear	P
equivalent-circuit battery	property	no	monotone in charge	T
state of health	storage	no	linear	P
Plant equipment				
chiller power from load	property	yes	increasing	P
chiller staging	piecewise	yes	—	P
thermal storage mode	piecewise	yes	—	P

D.2 The four patterns in the engine

Carrier flow. $r = P - q\,\varepsilon$, the flow of one quantity carried by the flow q of another at intensity ε. *Jacobian*: analytic, $\{P : 1,\ q : -\varepsilon,\ \varepsilon : -q\}$. *Shape*: bilinear, so not convex in q and ε together; a census over both needs the conditions of §14.7. *Source*: engine, `CarrierFlow`.

Gradient flow. $r = P - g(\phi_i, \phi_j; \theta)$, with g any function of the two potentials. *Jacobian*: analytic if the modeler supplies it, otherwise a central difference with step $10^{-6}\max(1, |\phi|)$. *Kinks*: not detected; whatever g has. *Source*: engine, `GradientFlow`.

Property. $r = y - f(x_1, \ldots, x_k)$, a relation among variables at one place. *Jacobian*: analytic if supplied, otherwise a central difference. When the Jacobian is differenced the engine probes for a kink at every evaluation: it compares the second difference with the first, and warns when their ratio exceeds 0.1. *Source*: engine, `PropertyClosure`.

Piecewise. $r = y - f_b(x)$ on the active branch b. Each branch may carry a guard, a margin that is non-negative where the branch is admissible. The engine solves with the branches frozen, then asks each closure which branch its guards now prefer. The current branch stays while it is admissible, and otherwise the first admissible branch in construction order is taken, so overlapping guards give hysteresis. The loop repeats at most eight times, and a closure that flips more than three times is relaxed and reported as a failure to converge. This is the regime loop of §7.3; the probabilistic model instead weighs the branches (Proposition 7.2). *Source*: engine, `PiecewiseClosure`, and `numerics/regimes`.

D.3 Thermal

Conductance.

$$r = Q - UA\,(T_h - T_c).$$

Pattern: gradient. *Unknowns*: Q, T_h, T_c. *Parameters*: UA [W/K]. *Jacobian*: analytic, constant. *Shape*: linear. *Prior*: UA, which is seldom known well at the start and drifts as surfaces foul. *Source*: `thermo`, `sensible_heat`; it is equation (1.5).

Series conduction.

$$r = Q - \frac{T_h - T_c}{\sum_i R_i}.$$

Pattern: gradient. *Parameters*: layer resistances R_i [K/W]. *Jacobian*: analytic, constant, $\mp 1/\sum_i R_i$ on the temperatures. *Shape*: linear. *Prior*: the least known layer, often a contact resistance; the sum is identifiable, the layers separately are not (§12.7). *Source*: `thermo`, `SeriesConduction`.

Convection with a correlation.

$$r = Q - hA\,(T_s - T_\infty), \qquad h = \frac{k}{L}\,C\,\mathrm{Re}^m\,\mathrm{Pr}^n.$$

Pattern: gradient. *Unknowns*: Q, T_s, T_∞, and the velocity inside Re when it is solved for. *Parameters*: A, the length L, the fluid properties, and the correlation constants C, m, n. *Kinks*:

where the correlation changes, at the laminar-to-turbulent transition, C, m and n jump. *Shape*: linear in the temperature difference at fixed velocity, increasing and concave in velocity. *Prior*: C, since empirical correlations carry errors that can reach about twenty-five percent [6]. *Source*: textbook [6].

Radiation between two surfaces.

$$r_i = q_i - k\,(T_i^4 - T_j^4) - G_i\,(T_i - T_{\mathrm{amb}}), \qquad r_j = q_j + k\,(T_i^4 - T_j^4) - G_j\,(T_j - T_{\mathrm{amb}}).$$

Pattern: custom, two residuals from one closure. *Unknowns*: q_i, q_j, T_i, T_j. *Parameters*: $k = \sigma\,\epsilon\,A$ with view factor [W/K^4], and room conductances G_i, G_j [W/K]. *Jacobian*: analytic, $4kT^3$ on each temperature. *Shape*: increasing in the hotter temperature, not convex over both. *Prior*: the emissivity inside k, which depends on the surface's condition. *Source*: `thermo`, `RadiativeCouple2Node`; the law is in [6].

Counterflow exchanger, fixed capacity rates.

$$r = Q - \varepsilon\,C_{\min}\,(T_{h,\mathrm{in}} - T_{c,\mathrm{in}}), \quad \varepsilon = \frac{1-e}{1-C_r e},\ e = \exp[-\mathrm{NTU}(1-C_r)],$$

with $\mathrm{NTU} = UA/C_{\min}$, $C_r = C_{\min}/C_{\max}$, and $\varepsilon = \mathrm{NTU}/(1+\mathrm{NTU})$ when $C_r = 1$. *Pattern*: custom. *Parameters*: UA, C_h, C_c [W/K], so ε is fixed when the closure is built. *Jacobian*: analytic, constant, $\mp\varepsilon C_{\min}$. *Shape*: linear in the inlet temperatures; the product $\varepsilon C_{\min}$ increases with UA. *Prior*: UA, for fouling. *Source*: `thermo`, `HeatExchangerEpsNTU`; the effectiveness relation is in [42].

Counterflow exchanger, variable capacity rates. The same residual with C_h and C_c as unknowns, so the exchanger responds to its flows. *Jacobian*: analytic in the temperatures; the capacity entries are differenced inside the closure. *Kinks*: at $C_h = C_c$, where $C_{\min}$ and $C_{\max}$ swap and the effectiveness formula changes. The closure refuses a capacity rate at or below a floor. *Shape*: not evident from the relation. *Source*: `thermo`, `HeatExchangerEpsNTUVariable`.

Cooling tower.

$$r = Q - \varepsilon_{\mathrm{eff}}(a)\,C_{cw}\,\max(0,\,T_{cw} - T_{wb}), \qquad \varepsilon_{\mathrm{eff}}(a) = \varepsilon_d\,\min(1, \max(0, a)),$$

with a the air-flow ratio when it is modeled, and $\varepsilon_{\mathrm{eff}} = \varepsilon_d$ when it is not. *Pattern*: gradient. *Unknowns*: Q, the water temperature to the tower T_{cw}, the wet-bulb temperature T_{wb}, and a. *Parameters*: C_{cw} [W/K], design effectiveness $\varepsilon_d \in [0, 1]$. *Kinks*: at $T_{cw} = T_{wb}$, $a = 0$ and $a = 1$. *Shape*: non-decreasing in both the approach and the air flow. *Prior*: ε_d, which is a fit to a manufacturer's curve. *Source*: `thermo`, `CoolingTowerEffectiveness`.

Waterside economizer.

$$r = Q - k\,\max(0,\,T_{chw,r} - T_{cw,s}), \qquad k = \varepsilon\,C_{\min}.$$

Pattern: property. *Kinks*: at the crossover $T_{chw,r} = T_{cw,s}$, below which the economizer does nothing and the Jacobian is zero. *Shape*: non-decreasing and convex in the temperature difference. *Source*: `datacenter`, condenser loop.

Enthalpy carried by mass; throttle.

$$r = P - \dot{m}\, h, \qquad r = h_{\text{out}} - h_{\text{in}}.$$

Pattern: carrier; custom. *Shape*: bilinear; linear. The throttle is exact: an adiabatic valve conserves enthalpy. *Source*: `thermo`, `enthalpy_advection` and `Throttle`; the duct heater, `make_duct_heater`, is two carrier edges around a node with a source and a loss.

Temperature-dependent load.

$$r = P - P_{\text{base}} - \alpha\,(T - T_{\text{ref}}).$$

Pattern: property. *Parameters*: α [W/K], T_{ref} [K]; P_{base} may be a schedule. *Shape*: linear. *Prior*: α, a calibrated fan-and-leakage slope. *Source*: `datacenter`, rack load.

D.4 Fluid

Quadratic resistance; damper.

$$r = (p_i - p_j) - k\,\dot{m}\,|\dot{m}| - \epsilon\,\dot{m}, \qquad k(\theta) = k_{\text{open}} / \max(\theta, \theta_{\min})^2.$$

Pattern: gradient. *Unknowns*: $\dot{m}$, p_i, p_j, and the damper position θ. *Parameters*: k [Pa/(kg/s)^2]; an optional linear term ϵ. *Jacobian*: analytic, $-2k|\dot{m}| - \epsilon$ in $\dot{m}$, which is zero at no flow unless $\epsilon > 0$; that is what the linear term is for. *Kinks*: the second derivative jumps at $\dot{m} = 0$; the damper kinks at $\theta_{\min}$. *Shape*: increasing and odd in $\dot{m}$. *Prior*: k, which is set by fittings nobody measured. *Source*: `thermo`, `FlowResistance` and `DamperResistance`.

Water main, Hazen–Williams.

$$r = (h_i - h_j) - K_e\, Q_e\, (Q_e^2 + q_0^2)^{0.426}, \qquad K_e = \frac{10.667\, L_e}{C_e^{1.852}\, D_e^{4.871}}.$$

Pattern: gradient. *Parameters*: length L [m], diameter D [m], roughness coefficient C (about 130 for new ductile iron, falling towards 80 as a main ages). *Kinks*: none, but the unregularized law $\Delta h = KQ^{1.852}$ has infinite inverse slope at zero flow; the reference flow q_0 above removes it, at a relative cost of $0.426\,(q_0/Q)^2$, and makes the law odd so that reversal needs no case split. *Shape*: increasing, between linear and quadratic; the curvature is largest near zero flow, which is where a night-time leak search works. *Prior*: C, which no utility has measured since the main was laid; Chapter 22 shows what a ten percent error in one of them does, and a companion study on a laboratory rig infers thirty-six such parameters at once. *Source*: Williams and Hazen [100]; the form used here is EPANET's [79].

Pipe friction.

$$r = (p_i - p_j) - f(\text{Re},\, e/D)\, \frac{L}{D}\, \frac{\rho v^2}{2}, \qquad f = 64/\text{Re (laminar)},\ \text{Colebrook (turbulent)}.$$

Pattern: gradient. *Parameters*: length L, diameter D, roughness e, density ρ, viscosity. *Kinks*: at the transition near $\text{Re} \approx 2300$ the friction factor jumps; a smooth blend is the usual remedy. *Shape*: increasing in flow; between linear (laminar) and quadratic (fully rough). *Prior*: the roughness e, which grows with age. *Source*: textbook [99].

Laminar pipe.

$$r = \dot{V} - \frac{\pi D^4}{128\,\mu L}\,(p_i - p_j).$$

Pattern: gradient. *Shape*: linear, the fluid analogue of a conductance. *Prior*: the viscosity μ, through its temperature dependence. *Source*: textbook [99].

Fan or pump curve.

$$r = (p_i - p_j) - \left(h_0\, s^2 - h_2\, \dot{m}\,|\dot{m}|\right),$$

with p_i and p_j the pressures at the edge's two ends in the plugin's orientation, so that the curve's head is the pressure difference along the edge. *Pattern*: gradient. *Unknowns*: $\dot{m}$, the two pressures, and the speed ratio s. *Parameters*: shutoff head h_0 [Pa], curvature h_2 [Pa/(kg/s)^2]. *Jacobian*: analytic, $+2h_2|\dot{m}|$ in $\dot{m}$ and $-2h_0 s$ in s. *Kinks*: C^1 at no flow. *Shape*: head decreasing in flow and increasing in speed squared, the affinity laws at fixed curve shape. *Prior*: h_0 and h_2, which wear moves. *Source*: `thermo`, `FanCurve`.

Fan and pump power.

$$r = P - P_{\text{rated}}\,(\dot{m}/\dot{m}_{\text{rated}})^3 \quad \text{or} \quad r = P - P_{\text{nom}}\,u^3.$$

Pattern: property. *Shape*: convex and increasing, the cube law of the affinity relations. *Prior*: the rated power, since the cube law holds only near the design point. *Source*: `thermo`, `pump_work`; `hvac`, fan power.

Valve with a loss coefficient.

$$r = (p_i - p_j) - K(\theta)\,\frac{\rho v^2}{2}.$$

Pattern: gradient. *Unknowns*: the flow, the pressures, the stem position θ. *Parameters*: the characteristic $K(\theta)$, linear or equal-percentage in the opening. *Kinks*: at full closure, where $K \to \infty$; a floor on the opening keeps the residual finite. *Shape*: increasing in flow, decreasing in opening. *Source*: textbook [99]; the damper above is the same relation with $K \propto 1/\theta^2$.

Check valve. Two branches: open, $r = (p_i - p_j) - k\,\dot{m}|\dot{m}|$ with guard $\dot{m} \geq 0$; closed, $r = \dot{m}$ with guard $p_j - p_i \geq 0$. *Pattern*: piecewise. *Kinks*: the branch switch is the whole point. *Shape*: monotone. The regime loop settles it in one or two passes; the probabilistic model weighs the two branches when the pressure difference is uncertain (§7.2). *Source*: textbook; any valve in [99] with a one-way guard.

Isothermal gas pipe.

$$r = (p_i^2 - p_j^2) - K\,\dot{m}\,|\dot{m}|.$$

Pattern: gradient, with the square of pressure as the potential. *Parameters*: K, from length, diameter, friction factor, temperature and the gas constant. *Kinks*: C^1 at no flow. *Shape*: increasing in flow. The squared potential makes the relation linear in p^2 for a fixed flow, which is why gas networks are usually solved in p^2. *Prior*: the friction factor inside K. *Source*: textbook [99].

Ideal-gas mass flow.

$$r = \dot{m} - \frac{p}{R\,T}\,\dot{V}.$$

Pattern: property, with the density fixed when the closure is built. *Shape*: linear in $\dot{V}$. *Prior*: the volume flow, which the code's own documentation suggests pinning or giving a prior. *Source*: `thermo`, `ideal_gas_mass_flow`.

Mixing of two streams.

$$r = T_{\mathrm{mix}} - f\,T_{oa} - (1-f)\,T_{ra}, \qquad r = \dot{m}_{oa} - f\,\dot{m}_{\mathrm{supply}}.$$

Pattern: property. *Shape*: bilinear in the fraction f and the temperatures. *Prior*: f, the outdoor-air fraction, which dampers set and few sensors report; a carbon-dioxide balance on the zone, $G = (\dot{m}_{oa}/\rho)(C_{\mathrm{zone}} - C_{oa})$, makes it identifiable. *Source*: `hvac`, mixed-air closures.

D.5 Moist air

Humidity ratio and moist-air enthalpy.

$$W = 0.62198\,\frac{p_w}{p - p_w}, \qquad h = 1006\,t + W\,(2.501 \times 10^6 + 1860\,t),$$

with t in degrees Celsius and the saturation pressure from the Magnus form

$$p_{ws} = 0.61094\,\exp\!\left[\frac{17.625\,t}{t + 243.04}\right]\ \mathrm{kPa}.$$

Pattern: property, when written as closures. *Shape*: increasing in p_w and in t. *Source*: `thermo`, the functions of `psychro`, which the plugins call inside other closures; the relations are in [4].

Coil condensation.

$$r = \dot{m}_{\mathrm{cond}} - \dot{m}_{\mathrm{air}}\,(1 - \mathrm{BF})\,\max\!\big(0,\, W_{\mathrm{zone}} - W_{\mathrm{adp}}\big),$$

with W_{adp} the saturated humidity ratio at the coil's apparatus dew point, the chilled-water temperature plus an approach. The latent duty is $\dot{m}_{\mathrm{cond}}$ times the heat of vaporization. *Pattern*: property. *Parameters*: bypass factor BF, approach [K]. *Jacobian*: differenced. *Kinks*: at $W_{\mathrm{zone}} = W_{\mathrm{adp}}$, where the coil starts to condense. *Shape*: non-decreasing in the zone's humidity. *Prior*: the bypass factor. *Source*: `datacenter`, the CRAH zone.

Moisture carried by mass.

$$r = \dot{m}_w - \dot{m}\,W.$$

Pattern: carrier. *Source*: `thermo`, `moisture_advection`.

D.6 Electrical

DC branch.

$$r = F - b\,(\theta_i - \theta_j), \qquad b = 1/X.$$

Pattern: gradient. *Parameters*: the susceptance b [p.u.]. *Jacobian*: analytic, $\mp b$. *Shape*: linear. *Prior*: b for a line whose impedance is in doubt; more often none, since the DC model's error is structural. *Source*: `electrical`, `dc_branch`; the approximation is [87].

AC branch in polar form. With $\delta = \theta_i - \theta_j$, series admittance $G + jB = 1/(R + jX)$, half line charging B_{sh} and tap t, the four end flows are

$$\begin{aligned} P_{\text{from}} &= V_i^2 G/t^2 - V_i V_j \,(G\cos\delta + B\sin\delta)/t, \\ Q_{\text{from}} &= -V_i^2 (B + B_{sh})/t^2 - V_i V_j \,(G\sin\delta - B\cos\delta)/t, \\ P_{\text{to}} &= V_j^2 G - V_i V_j \,(G\cos\delta - B\sin\delta)/t, \\ Q_{\text{to}} &= -V_j^2 (B + B_{sh}) + V_i V_j \,(G\sin\delta + B\cos\delta)/t, \end{aligned}$$

and each closure is $r = F - (1 + f)\, F(V, \theta)$ for its flow. *Pattern*: custom. *Unknowns*: the flow, both voltages and angles, and an optional latent admittance shift f. *Jacobian*: analytic; every partial is scaled by $1 + f$, and $\partial r/\partial f = -F$. *Shape*: trigonometric, neither monotone nor convex. *Prior*: f, a latent fault of the kind of Chapter 3, zero-mean with a small width. *Source*: `electrical`, `ACBranchPolar`; the equations are in [101]. Each line meets its two buses through a midpoint node that absorbs the loss $P_{\text{from}} + P_{\text{to}}$.

Transformer with a tap. The AC branch with $t \neq 1$. It carries no latent fault in the current plugin. *Source*: `electrical`, `transformer_polar`; `power_flow`, `attach_ac_line` with a tap.

Slack and voltage-controlled buses. These are pins, not closures: $r = \theta - 0$ and $r = V - V_{\text{set}}$ at the slack, $r = V - V_{\text{set}}$ at a voltage-controlled bus. The generator's injections are free flows from a reservoir, closed by the bus balances alone. A pinned variable stays in the list of unknowns, so a study can observe it or put a prior on it. *Source*: `electrical`, `bus_residuals`; `power_flow`, `attach_bus`.

Reactive limits of a generator. An outer loop, not a closure. After a solve, a voltage-controlled bus whose generator exceeds its reactive band is rewritten as a load bus with the reactive output held at the violated limit, and the model is solved once more. There is one such pass, and no return to voltage control. *Kinks*: the switch is a kink in every quantity downstream. *Source*: `power_flow`, `solve_with_q_limits`.

Transformer loss.

$$r = L - L_0 - L_{cu}\,\frac{(\sum g)^2}{S^2}.$$

Pattern: property. *Parameters*: no-load loss L_0, copper loss at rating L_{cu}, rating S [kW]. *Shape*: convex in the load $\sum g$. *Prior*: L_{cu}, a calibrated coefficient. *Source*: `facility`, `TransformerLoss`.

UPS loss.

$$r = L - \ell_f \sum g - \ell_q\,\frac{(\sum g)^2}{S} - L_0.$$

Pattern: property. *Shape*: convex. *Prior*: the coefficients ℓ_f, ℓ_q, calibrated to an efficiency curve. *Source*: `facility`, `UpsLoss`.

PDU loss.

$$r = L - L_0 - p \sum g.$$

Pattern: property. *Shape*: linear. *Source*: `facility`, `PduLoss`.

Fixed share of a flow.

$$r = s_{\text{ref}}\, F - s_F\, F_{\text{ref}}.$$

Pattern: custom. *Shape*: linear. Written without dividing by s_{ref}, so the residual is well defined for any share. *Source*: `facility`, `SplitShare`.

Generator fuel curve.

$$r = \dot{V}_{\text{fuel}} - (a + b\,\lambda)\, g,$$

with λ the load fraction and $g \in \{0, 1\}$ the run state. *Pattern*: property. *Shape*: affine and non-decreasing in λ. *Kinks*: the start and stop are held outside the solver: g is set per segment by a switching driver, so each solve sees a smooth relation. *Source*: `facility`, `GensetFuelCurve`.

D.7 Storage and electrochemistry

Battery energy with efficiencies.

$$\frac{\mathrm{d}E}{\mathrm{d}t} = \eta_c P_c - P_d/\eta_d, \qquad L = (1-\eta_c)P_c + (1/\eta_d - 1)P_d, \qquad \text{SoC} = E/E_{\text{cap}}.$$

Pattern: a storage node for the energy, with property closures for the loss and the state of charge. *Parameters*: the efficiencies η_c, η_d, and the capacity. *Shape*: linear. A steady solve must pin the energy or the state of charge, since a storage node in steady state has no row of its own (§16.1). *Prior*: the efficiencies. *Source*: `battery_storage`, `attach_battery_pack`. The plugin's pack is this bookkeeping model; it has no voltage curve and no internal resistance.

Signed power port.

$$P_c = \max(-s, 0), \qquad P_d = \max(s, 0).$$

Pattern: property. *Kinks*: at $s = 0$. The kink is harmless only because the setpoint s is pinned and never a Newton unknown; if a study makes s uncertain, this closure is where the Laplace approximation fails. *Source*: `battery_storage`, power port.

Equivalent-circuit battery.

$$r = V - \text{OCV}(\text{SoC}) + R_0\, I,$$

with further resistor–capacitor pairs for the slow voltage response. *Pattern*: property. *Unknowns*: terminal voltage, current, state of charge. *Parameters*: the open-circuit voltage curve, the series resistance R_0. *Shape*: OCV increasing in the state of charge, flat in the middle of the range, which makes the state of charge weakly identifiable from voltage there. *Prior*: R_0 and the capacity, both of which age. *Source*: textbook [73]; not in the plugin.

State of health.

$$\frac{\mathrm{d}D}{\mathrm{d}t} = \dot{a}, \qquad \text{SoH} = \text{SoH}_0 - D, \qquad C_{\text{eff}} = C_{\text{nom}}\,\text{SoH},$$

with an aging rate $\dot{a}$ from throughput or from an Arrhenius calendar law,

$$\dot{a} = \dot{a}_{\text{ref}}\, \exp\!\left[\frac{E_a}{R}\left(\frac{1}{T_{\text{ref}}} - \frac{1}{T}\right)\right].$$

Pattern: a storage node for the accumulated damage D, with property closures. *Shape*: linear in D; the capacity relation is bilinear. *Prior*: the reference rate $\dot{a}_{\mathrm{ref}}$, the least known number in battery aging. *Source*: `battery_storage`, `attach_soh_evolution`.

D.8 Plant equipment

Chiller power from load.

$$r = P - \frac{Q}{\mathrm{COP}}, \qquad \mathrm{COP} = \max\!\big(\mathrm{COP}_{\min},\ \mathrm{COP}_{\mathrm{pk}} - c\,(x - x_{\mathrm{opt}})^2\big),$$

with x the load per running unit as a fraction of one unit's capacity, and $P = 0$ at no load. *Pattern*: property. *Jacobian*: differenced. *Kinks*: at zero load, at the COP floor, and wherever the stage count changes, since the stage enters through rounding. *Shape*: increasing in load at fixed stage for the default curve. A condenser-loop variant also derates the COP linearly with the condenser-water temperature. *Prior*: the peak COP and the curvature, fitted to a part-load curve. *Source*: `datacenter`, `ChillerBankPowerClosure` and the condenser loop.

Chiller staging. A piecewise closure with one branch per stage, $r = \mathrm{stage} - k$. Stage k is admissible between $(k-1)$ units at their stage-down fraction and k units at their stage-up fraction, 0.70 and 0.90 of capacity by default. *Kinks*: every stage change. *Shape*: the admissible bands of neighbouring stages overlap, and since the current stage is kept while admissible, the overlap is a deadband that stops the plant from cycling. *Source*: `datacenter`, `ChillerBankStageClosure`.

Thermal storage mode. A piecewise closure with branches charge, hold and discharge, $r = \mathrm{mode} - m$ with $m \in \{+1, 0, -1\}$. Guards combine the plant load with the tank's state of charge: charge below a load threshold while the tank is not full, discharge above another while it is not empty, hold otherwise. *Kinks*: every mode change. *Source*: `datacenter`, `TesModeClosure`.

Bibliography

[1] Ali Abur and Antonio Gomez Exposito. *Power System State Estimation: Theory and Implementation*. Marcel Dekker, 2004.

[2] Athanasios C. Antoulas. *Approximation of Large-Scale Dynamical Systems*. SIAM, Philadelphia, 2005.

[3] Stefan Arnborg, Derek G. Corneil, and Andrzej Proskurowski. Complexity of finding embeddings in a k-tree. *SIAM Journal on Algebraic and Discrete Methods*, 8(2):277–284, 1987.

[4] ASHRAE. *ASHRAE Handbook—Fundamentals*. ASHRAE, Atlanta, 2021.

[5] Albert Benveniste, Benoît Caillaud, and Mathias Malandain. The mathematical foundations of physical systems modeling languages. *Annual Reviews in Control*, 50:72–118, 2020.

[6] Theodore L. Bergman, Adrienne S. Lavine, Frank P. Incropera, and David P. DeWitt. *Fundamentals of Heat and Mass Transfer*. Wiley, 7th edition, 2011.

[7] Julian Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society: Series B (Methodological)*, 36(2):192–236, 1974.

[8] Roy Billinton and Ronald N. Allan. *Reliability Evaluation of Power Systems*. Plenum Press, New York, 2nd edition, 1996.

[9] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[10] José Luis Blanco-Claraco, Antonio Leanza, and Giulio Reina. A general framework for modeling and dynamic simulation of multibody systems using factor graphs. *Nonlinear Dynamics*, 105(3):2031–2053, 2021.

[11] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[12] K. E. Brenan, S. L. Campbell, and L. R. Petzold. *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*. SIAM, 1996.

[13] Kathryn Chaloner and Isabella Verdinelli. Bayesian experimental design: A review. *Statistical Science*, 10(3):273–304, 1995.

[14] Phani Chavali and Arye Nehorai. Distributed power system state estimation using factor graphs. *IEEE Transactions on Signal Processing*, 63(11):2864–2876, 2015.

[15] Roy R. Craig and Mervyn C. C. Bampton. Coupling of substructures for dynamic analyses. *AIAA Journal*, 6(7):1313–1319, 1968.

[16] Timothy A. Davis. *Direct Methods for Sparse Linear Systems*. SIAM, 2006.

[17] Frank Dellaert and Michael Kaess. Factor graphs for robot perception. *Foundations and Trends in Robotics*, 6(1–2):1–139, 2017.

[18] Florian Dörfler and Francesco Bullo. Kron reduction of graphs with applications to electrical networks. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 60(1):150–163, 2013.

[19] Iain S. Duff, Albert M. Erisman, and John K. Reid. *Direct Methods for Sparse Matrices*. Oxford University Press, 2nd edition, 2017.

[20] A. L. Dulmage and N. S. Mendelsohn. Coverings of bipartite graphs. *Canadian Journal of Mathematics*, 10:517–534, 1958.

[21] A. M. Erisman and W. F. Tinney. On computing certain elements of the inverse of a sparse matrix. *Communications of the ACM*, 18(3):177–179, 1975.

[22] Lawrence C. Evans and Ronald F. Gariepy. *Measure Theory and Fine Properties of Functions*. CRC Press, 1992.

[23] Herbert Federer. *Geometric Measure Theory*. Springer, 1969.

[24] Miroslav Fiedler. Algebraic connectivity of graphs. *Czechoslovak Mathematical Journal*, 23 (2):298–305, 1973.

[25] Alan George. Nested dissection of a regular finite element mesh. *SIAM Journal on Numerical Analysis*, 10(2):345–363, 1973.

[26] Claudio Gomes, Casper Thule, David Broman, Peter Gorm Larsen, and Hans Vangheluwe. Co-simulation: A survey. *ACM Computing Surveys*, 51(3):1–33, 2018.

[27] N. J. Gordon, D. J. Salmond, and A. F. M. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings F: Radar and Signal Processing*, 140 (2):107–113, 1993.

[28] Teoman Guler, George Gross, and Minghai Liu. Generalized line outage distribution factors. *IEEE Transactions on Power Systems*, 22(2):879–881, 2007.

[29] Kjell Gustafsson, Michael Lundh, and Gustaf Söderlind. A PI stepsize control for the numerical solution of ordinary differential equations. *BIT Numerical Mathematics*, 28(2): 270–287, 1988.

[30] Robert J. Guyan. Reduction of stiffness and mass matrices. *AIAA Journal*, 3(2):380, 1965.

[31] William W. Hager. Updating the inverse of a matrix. *SIAM Review*, 31(2):221–239, 1989.

[32] Ernst Hairer and Gerhard Wanner. *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems*. Springer, 2nd edition, 1996.

[33] Qing Han, Ronald T. Eguchi, Sharad Mehrotra, and Nalini Venkatasubramanian. Enabling state estimation for fault identification in water distribution systems under large disasters. In *Proceedings of the 37th IEEE Symposium on Reliable Distributed Systems*, pages 161–170, 2018.

[34] Nicholas J. Higham. *Accuracy and Stability of Numerical Algorithms*. SIAM, 2nd edition, 2002.

[35] Paul Irofti, Luis Romero-Ben, Florin Stoican, and Vicenç Puig. Factor graph optimization for leak localization in water distribution networks. arXiv:2509.10982, 2025.

[36] Simon J. Julier and Jeffrey K. Uhlmann. A non-divergent estimation algorithm in the presence of unknown correlations. In *Proceedings of the American Control Conference*, volume 4, pages 2369–2373, 1997.

[37] Jari Kaipio and Erkki Somersalo. *Statistical and Computational Inverse Problems*. Springer, 2005.

[38] R. E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45, 1960.

[39] Dean C. Karnopp and Ronald C. Rosenberg. *Analysis and Simulation of Multiport Systems: The Bond Graph Approach to Physical System Dynamics*. MIT Press, Cambridge, MA, 1968.

[40] George Karypis and Vipin Kumar. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM Journal on Scientific Computing*, 20(1):359–392, 1998.

[41] Robert E. Kass and Adrian E. Raftery. Bayes factors. *Journal of the American Statistical Association*, 90(430):773–795, 1995.

[42] William M. Kays and Alexander L. London. *Compact Heat Exchangers*. McGraw-Hill, 3rd edition, 1984.

[43] J. E. Kelley. The cutting-plane method for solving convex programs. *Journal of the Society for Industrial and Applied Mathematics*, 8(4):703–712, 1960.

[44] Brian W. Kernighan and Shen Lin. An efficient heuristic procedure for partitioning graphs. *Bell System Technical Journal*, 49(2):291–307, 1970.

[45] Daphne Koller and Nir Friedman. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009.

[46] Gabriel Kron. *Tensor Analysis of Networks*. Wiley, New York, 1939.

[47] Frank R. Kschischang, Brendan J. Frey, and Hans-Andrea Loeliger. Factor graphs and the sum-product algorithm. *IEEE Transactions on Information Theory*, 47(2):498–519, 2001.

[48] Steffen L. Lauritzen. Propagation of probabilities, means, and variances in mixed graphical association models. *Journal of the American Statistical Association*, 87(420):1098–1108, 1992.

[49] Steffen L. Lauritzen. *Graphical Models*. Oxford University Press, 1996.

[50] Steffen L. Lauritzen and David J. Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society: Series B (Methodological)*, 50(2):157–224, 1988.

[51] Steffen L. Lauritzen and Nanny Wermuth. Graphical models for associations between variables, some of which are qualitative and some quantitative. *Annals of Statistics*, 17(1): 31–57, 1989.

[52] Dennis V. Lindley. On a measure of the information provided by an experiment. *Annals of Mathematical Statistics*, 27(4):986–1005, 1956.

[53] Richard J. Lipton and Robert Endre Tarjan. A separator theorem for planar graphs. *SIAM Journal on Applied Mathematics*, 36(2):177–189, 1979.

[54] C. H. Lo, Y. K. Wong, and A. B. Rad. Bond graph based Bayesian network for fault diagnosis. *Applied Soft Computing*, 11(1):1208–1212, 2011.

[55] Hans-Andrea Loeliger. An introduction to factor graphs. *IEEE Signal Processing Magazine*, 21(1):28–41, 2004.

[56] David J. C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003.

[57] Richard S. H. Mah, Gregory M. Stanley, and Dennis M. Downing. Reconciliation and rectification of process flow and inventory data. *Industrial & Engineering Chemistry Process Design and Development*, 15(1):175–183, 1976.

[58] Dmitry M. Malioutov, Jason K. Johnson, and Alan S. Willsky. Walk-sums and belief propagation in Gaussian graphical models. *Journal of Machine Learning Research*, 7: 2031–2064, 2006.

[59] Albert W. Marshall and Ingram Olkin. A multivariate exponential distribution. *Journal of the American Statistical Association*, 62(317):30–44, 1967.

[60] Mihajlo D. Mesarović, D. Macko, and Yasuhiko Takahara. *Theory of Hierarchical, Multilevel, Systems*. Academic Press, New York, 1970.

[61] Gary L. Miller, Shang-Hua Teng, William Thurston, and Stephen A. Vavasis. Geometric separators for finite-element meshes. *SIAM Journal on Scientific Computing*, 19(2):364–386, 1998.

[62] Thomas P. Minka. Expectation propagation for approximate Bayesian inference. In *Proceedings of the 17th Conference on Uncertainty in Artificial Intelligence*, pages 362–369, 2001.

[63] Modelica Association. Modelica: A unified object-oriented language for systems modeling. language specification, version 3.6. `https://specification.modelica.org`, 2023.

[64] Alcir Monticelli. *State Estimation in Electric Power Systems: A Generalized Approach*. Kluwer Academic, Boston, 1999.

[65] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, 2nd edition, 2006.

[66] George Oster, Alan Perelson, and Aharon Katchalsky. Network thermodynamics. *Nature*, 234:393–399, 1971.

[67] Art B. Owen. Sobol' indices and Shapley value. *SIAM/ASA Journal on Uncertainty Quantification*, 2(1):245–251, 2014.

[68] Constantinos C. Pantelides. The consistent initialization of differential-algebraic systems. *SIAM Journal on Scientific and Statistical Computing*, 9(2):213–231, 1988.

[69] Henry M. Paynter. *Analysis and Design of Engineering Systems*. MIT Press, Cambridge, MA, 1961.

[70] Judea Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2nd edition, 2009.

[71] Ramon Pérez, Vicenç Puig, Josep Pascual, Joseba Quevedo, Edson Landeros, and Antonio Peralta. Methodology for leakage isolation using pressure sensitivity analysis in water distribution networks. *Control Engineering Practice*, 19(10):1157–1167, 2011. doi: 10.1016/j.conengprac.2011.06.004.

[72] Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of Causal Inference: Foundations and Learning Algorithms*. MIT Press, 2017.

[73] Gregory L. Plett. *Battery Management Systems, Volume I: Battery Modeling*. Artech House, 2015.

[74] Alex Pothen, Horst D. Simon, and Kang-Pu Liou. Partitioning sparse matrices with eigenvectors of graphs. *SIAM Journal on Matrix Analysis and Applications*, 11(3):430–452, 1990.

[75] H. E. Rauch, F. Tung, and C. T. Striebel. Maximum likelihood estimates of linear dynamic systems. *AIAA Journal*, 3(8):1445–1450, 1965.

[76] Christian P. Robert and George Casella. *Monte Carlo Statistical Methods*. Springer, 2nd edition, 2004.

[77] R. Tyrrell Rockafellar and Stanislav Uryasev. Optimization of conditional value-at-risk. *Journal of Risk*, 2(3):21–41, 2000.

[78] Donald J. Rose, R. Endre Tarjan, and George S. Lueker. Algorithmic aspects of vertex elimination on graphs. *SIAM Journal on Computing*, 5(2):266–283, 1976.

[79] Lewis A. Rossman. EPANET 2 users manual. Technical Report EPA/600/R-00/057, United States Environmental Protection Agency, Cincinnati, Ohio, 2000.

[80] Håvard Rue and Leonhard Held. *Gaussian Markov Random Fields: Theory and Applications*. Chapman and Hall/CRC, 2005.

[81] Andrea Saltelli. Making best use of model evaluations to compute sensitivity indices. *Computer Physics Communications*, 145(2):280–297, 2002.

[82] Simo Särkkä and Lennart Svensson. *Bayesian Filtering and Smoothing*. Cambridge University Press, 2nd edition, 2023.

[83] Fred C. Schweppe and J. Wildes. Power system static-state estimation, part I: Exact model. *IEEE Transactions on Power Apparatus and Systems*, PAS-89(1):120–125, 1970.

[84] Ori Shental, Paul H. Siegel, Jack K. Wolf, Danny Bickson, and Danny Dolev. Gaussian belief propagation solver for systems of linear equations. In *Proceedings of the IEEE International Symposium on Information Theory*, pages 1863–1867, 2008.

[85] Ilya M. Sobol'. Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. *Mathematics and Computers in Simulation*, 55(1–3):271–280, 2001.

[86] Eunhye Song, Barry L. Nelson, and Jeremy Staum. Shapley effects for global sensitivity analysis: Theory and computation. *SIAM/ASA Journal on Uncertainty Quantification*, 4 (1):1060–1083, 2016.

[87] Brian Stott, Jorge Jardim, and Ongun Alsaç. DC power flow revisited. *IEEE Transactions on Power Systems*, 24(3):1290–1300, 2009.

[88] Andrew M. Stuart. Inverse problems: A Bayesian perspective. *Acta Numerica*, 19:451–559, 2010.

[89] Kaarthik Sundar, Carleton Coffrin, Harsha Nagarajan, and Russell Bent. Probabilistic N-k failure-identification for power systems. *Networks*, 71(3):302–321, 2018.

[90] Kaarthik Sundar, Andrew Mastin, Manuel Garcia, Russell Bent, and Jean-Paul Watson. Exact and heuristic approaches for the stochastic N-k interdiction in power grids. arXiv:2402.00217, 2024.

[91] Luke Tierney and Joseph B. Kadane. Accurate approximations for posterior moments and marginal densities. *Journal of the American Statistical Association*, 81(393):82–86, 1986.

[92] William F. Tinney and John W. Walker. Direct solutions of sparse network equations by optimally ordered triangular factorization. *Proceedings of the IEEE*, 55(11):1801–1809, 1967.

[93] Ezio Todini and S. Pilati. A gradient algorithm for the analysis of pipe networks. *Computer Applications in Water Supply*, 1:1–20, 1988.

[94] Andrea Toselli and Olof Widlund. *Domain Decomposition Methods: Algorithms and Theory*. Springer, 2005.

[95] Arjan J. van der Schaft and Bernhard M. Maschke. Port-Hamiltonian systems on graphs. *SIAM Journal on Control and Optimization*, 51(2):906–937, 2013.

[96] Martin J. Wainwright and Michael I. Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1(1–2):1–305, 2008.

[97] J. B. Ward. Equivalent circuits for power-flow studies. *Transactions of the American Institute of Electrical Engineers*, 68(1):373–382, 1949.

[98] Yair Weiss and William T. Freeman. Correctness of belief propagation in Gaussian graphical models of arbitrary topology. *Neural Computation*, 13(10):2173–2200, 2001.

[99] Frank M. White. *Fluid Mechanics*. McGraw-Hill Education, 8th edition, 2016.

[100] Gardner S. Williams and Allen Hazen. *Hydraulic Tables*. John Wiley and Sons, New York, 1905.

[101] Allen J. Wood, Bruce F. Wollenberg, and Gerald B. Sheblé. *Power Generation, Operation, and Control*. Wiley, 3rd edition, 2013.

[102] Mihalis Yannakakis. Computing the minimum fill-in is NP-complete. *SIAM Journal on Algebraic and Discrete Methods*, 2(1):77–79, 1981.