Sequential updating of a Bayesian thermal fault diagnosis in a data-center building from temperature readings that arrive one at a time

Ricardo Marquez, Ph.D.
rmarquez123@gmail.com

September 2026

Abstract

A thermal diagnosis of a data-center building estimates heat that no sensor measures directly from temperatures that sensors report one at a time. This report asks whether keeping such a diagnosis current requires solving the whole estimation problem again for every new reading. It is an inference study. A building of 4 rooms with 64 racks each (256 racks, 1,039 unknowns) carries an unknown heat term at every rack, every room and the building, and every link of its thermal network is sized for the heat it carries. The diagnosis is the Laplace approximation of the posterior over those terms given a baseline snapshot of 261 temperature readings, and 20 further rack readings arrive one at a time. Two workflows process the same stream: a full maximum-a-posteriori solve after every reading, and a rank-one update of the posterior mean and information matrix. After the stream the two agree to within 10^{-6} of every heat term's posterior standard deviation at every building size from 16 to 512 racks; the standard deviations range from 1.28 W for a rack to 1,640 W for the building. Measured against the largest move the stream made instead, the agreement holds only at the smallest building. On one workstation, measured on a quiet machine, the update was at least 9× (9–25× over five repeats) cheaper than the full solve on every reading at 256 racks, and its median at least 13× (13–21× over five repeats) cheaper at every size. The full solve costs one maximum-a-posteriori solve per reading; the update costs one such solve in total and, per reading, one sparse factorization and one back-substitution.

1 Introduction

Data centers consume a small but significant share of the world's electricity, and a large part of it is spent removing heat. Masanet, Shehabi, Lei, Smith and Koomey (2020) put data-center electricity use at about one percent of global demand and attribute its slow growth, while computing grew many times over, to efficiency gains including those in cooling; Barroso, Hölzle and Ranganathan (2019) describe how warehouse-scale facilities are cooled and how the overhead is reported. Cooling works only as well as operators understand where heat is produced and where it goes. A blocked air path, a failing fan or a server running a heavier load than planned shows up as heat the thermal plan does not account for, and a building has temperature sensors at racks, in rooms and on its water loops that could reveal it. Turning those temperatures into an estimate of the unaccounted heat at each rack, room and plant is an inference problem on a thermal network, and it has a practical difficulty: the readings do not arrive together. They arrive as a stream, one sensor at a time, and a diagnosis that is to be used for operation must follow that stream.

Folding one new measurement into an existing Gaussian estimate is among the oldest results in estimation. Kalman (1960) gave the recursive form for linear systems with Gaussian noise, and Särkkä (2013) presents it and its nonlinear extensions as sequential Bayesian updating; for a static unknown the recursion reduces to recursive least squares. Its algebra rests on the formula of Sherman and Morrison (1950) for the inverse of a matrix changed by a rank-one term, whose history and variants Hager (1989) reviews, and Davis and Hager (1999) show how to modify a sparse Cholesky factorization under such a change without refactoring it. On physical networks, Schweppe and Wildes (1970) framed the static state of a power system as a weighted least-squares estimate from redundant measurements, the construction still used to monitor transmission grids. Factor graphs (Kschischang, Frey and Loeliger 2001) write such problems as sparse products of local relations, and incremental smoothing exploits that structure when measurements arrive over time: Kaess, Ranganathan and Dellaert (2008) update a sparse QR factorization as measurements are added, Kaess et al. (2012) organize the update in a Bayes tree that also handles relinearization, and Dellaert and Kaess (2017) review the approach. The sparse LU factorization this study uses is SuperLU (Li 2005), called through SciPy (Virtanen et al. 2020). None of these constructions is new here, and none is claimed as new.

TERRA is a general-purpose engine for probabilistic inference on physical networks. It treats a network as a graph on which the physics is written locally: a conservation law at every node, a constitutive law on every edge, and, attached to the same graph, whatever the sensors report and whatever is known in advance. Each of these is a relation that holds only to within a stated tolerance, and the engine weighs them together to find the most probable state of the system, the uncertainty of every quantity in it, and the relative plausibility of competing explanations of the same readings. Domains enter through plugins that supply their laws, units and file formats; the inference and the bookkeeping are shared. This study uses its data-center plugin to write the heat balances of a building, to size its links and to attach an unknown heat term with its prior to every rack, room and the building, and it uses the engine's maximum-a-posteriori solve and its Laplace approximation as the reference against which a sequential update is compared. Section 3 states the framework and each of these constructs precisely.

The rank-one update is exact for a linear model with Gaussian noise, so what this report adds is a check rather than a method: whether, on a thermal network sized as real buildings are, an update carried out in floating point reaches the diagnosis that solving the whole problem again reaches, how that agreement should be judged, and what the two cost per reading over a 32-fold range of building sizes. The question is whether a new temperature reading can be absorbed into a building's thermal diagnosis without solving the estimation problem again. The construction runs both workflows on the same stream of readings at six building sizes and compares their posteriors and their wall-clock time. Section 2 describes the building and the data, Section 3 the model, the update and the metrics, Section 4 the results, Section 5 their limits, and Section 6 concludes.

2 System and data

The system is a data-center building of 4 rooms, drawn in Figure 1. Each room holds K racks that dissipate electrical power as heat. A rack's heat leaves through the air stream that crosses it into the room air; each room's air is cooled by a coil fed from the building's chilled-water loop, and the loop rejects the heat of all rooms to chilled water held at 290 K. A rack draws 300 W at a reference temperature of 300 K and 0.1 W more for every kelvin it runs above it, the leakage of its electronics. Every rack, every room and the building carries an unknown heat term, an extra heat source that the readings are allowed to reveal, and every one of them has a temperature sensor.

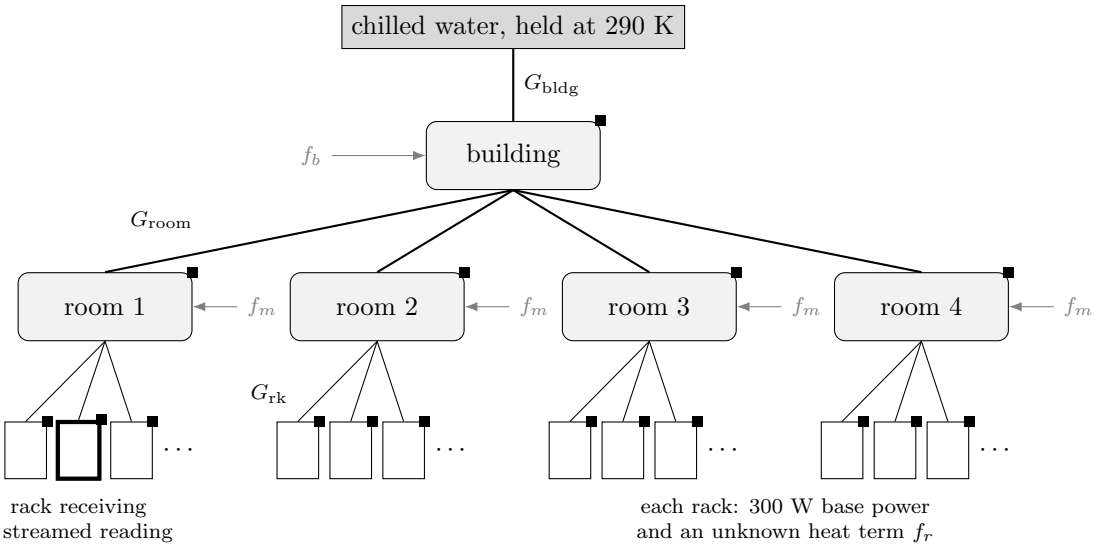


Figure 1: The building of this report. Heat produced in every rack flows through the rack's air stream to its room, through the room's cooling coil to the building's chilled-water loop and from there to chilled water held at 290 K. Every rack, every room and the building carries an unknown heat term (f_r, f_m, f_b), the quantities the diagnosis estimates, and every one of them has a temperature sensor (black squares). Three racks per room are drawn; the principal case has 64 per room, 256 in total. The heavy rack marks where one streamed reading lands: each new reading measures one rack, and the update must decide how far every other estimate in the building moves.

Every link is sized for the heat it carries at the design point. A rack's air stream is sized for a 12 K rise across the rack, and the room and building coils for approaches of 3 K and 3 K; a link's conductance is its design heat divided by its design temperature difference. Table 1 lists the sizes studied and the conductances they receive. At 128 racks per room a room rejects 38.4 kW and its coil needs 12,800 W/K. With these conductances every rack of the fault-free building runs at 308.05 K and every room at about 296.0 K, at every size.

No measured data enter the study. A forward solve of the building without any unknown heat supplies the true temperature at every sensor. The baseline snapshot is one reading of every sensor, every rack, every room and the building (261 readings in the principal case), and it gives the diagnosis the operator starts from. The stream is 20 further rack readings in the principal case and 5 per size in the sweep, cycling over the rooms and over racks in a fixed pattern. Each reading is its true value plus a deterministic perturbation of amplitude 0.02 K, a sine of the reading's index, standing in for sensor noise inside the 0.05 K noise level the model assumes. The sweep repeats the whole exercise at 6 sizes from 16 to 512 racks. Table 2 collects every input.

Figure 2 shows the model the diagnosis is built on, around one rack: the unknowns, the given base power, and the four kinds of rows that tie them together. The assumptions are these.

- The building is at steady state; nothing stores heat.
- Every relation is linear in the unknowns: heat flow on a link is its conductance times the temperature difference, and rack power is linear in rack temperature.
- Sensor noise is Gaussian, independent between readings, with standard deviation 0.05 K.

Table 1: The building sizes of the sweep, with 4 rooms at every size, and the conductances the sizing rule of Eqs. (8)–(10) gives them. Room heat is the design heat one room rejects. The rack conductance does not depend on size; the room and building conductances grow in proportion to the heat they carry. The principal case is the 64-racks-per-room building.

racks per room	racks	unknowns	room heat [kW]	G_{rk} [W/K]	G_{room} [W/K]	G_{bldg} [W/K]
4	16	79	1.2	25	400	1,600
8	32	143	2.4	25	800	3,200
16	64	271	4.8	25	1,600	6,400
32	128	527	9.6	25	3,200	12,800
64	256	1,039	19.2	25	6,400	25,600
128	512	2,063	38.4	25	12,800	51,200

Table 2: Every input of the study. The building, its loads and the readings are synthetic; everything else in the report is computed from these values.

quantity	value
rooms; racks per room (principal case)	4; 64
rack base power p_0; leakage slope α; reference T_{ref}	300 W; 0.1 W/K; 300 K
chilled water	290 K
design rack air rise; coil approaches	12 K; 3 K and 3 K
prior standard deviation of heat terms (rack / room / building)	200 / 800 / 2,000 W
sensor noise σ_s; pseudo-noise amplitude a	0.05 K; 0.02 K
physics tolerance σ_p	0.001
baseline readings (principal case)	261
streamed readings (principal case; per size in the sweep)	20; 5
pseudo-noise phases (baseline; stream)	7; 23
racks per room in the sweep	4, 8, 16, 32, 64, 128

- The unknown heat terms have zero-mean Gaussian priors with standard deviations 200 W (rack), 800 W (room) and 2,000 W (building).
- The physical relations are weighted with tolerance 0.001 W, so the diagnosis satisfies them to within that tolerance rather than exactly.
- Base powers, the leakage law, the conductances and the chilled-water temperature are known; only the heat terms are uncertain.

3 Methods

3.1 The thermal network

The unknowns are the temperature of every rack, room and the building, the power of every rack, the heat flow on every link and the heat term at every rack, room and the building; Table 6 in the appendix lists the symbols. For rack r in room m, energy balances at the rack, the rack's power follows its leakage law, and the heat on its link follows the link's conductance:

$$P_r + f_r - Q_r = 0, \tag{1}$$

$$P_r - p_0 - \alpha\,(T_r - T_{\mathrm{ref}}) = 0, \tag{2}$$

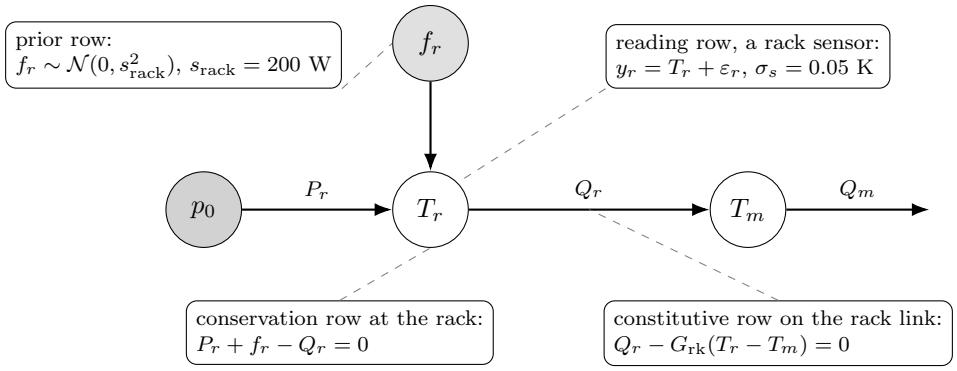


Figure 2: The model around one rack. White circles are unknowns that physics determines once the heat terms are fixed (the rack and room temperatures; the heat flows P_r, Q_r and Q_m are unknowns too), the shaded circle is the unknown heat term the diagnosis estimates, and the gray circle is the rack's base power, $p_0 = 300$ W, which is given. The boxes are the four kinds of rows attached to them: conservation at the rack, a constitutive law on its link to the room, a reading row for its sensor and a prior on the heat term. Rooms and the building carry the same four kinds of rows. Every row has a weight, and the diagnosis weighs all of them together.

$$Q_r - G_{\text{rk}}\,(T_r - T_m) = 0. \tag{3}$$

Room m collects the heat of its racks and passes it to the building, and the building passes the heat of all rooms to chilled water at the fixed temperature T_{chw}:

$$\textstyle\sum_{r \in m} Q_r + f_m - Q_m = 0, \tag{4}$$
$$Q_m - G_{\text{room}}\,(T_m - T_b) = 0, \tag{5}$$
$$\textstyle\sum_{m} Q_m + f_b - Q_b = 0, \tag{6}$$
$$Q_b - G_{\text{bldg}}\,(T_b - T_{\text{chw}}) = 0. \tag{7}$$

With K racks per room and $R = 4$ rooms, a design rack power p_0, a design air rise ΔT_{rise} of 12 K across a rack and a coil approach ΔT_{app} of 3 K, the sizing rule sets the three conductances to the design heat of each link over its design temperature difference:

$$G_{\text{rk}} = p_0 / \Delta T_{\text{rise}}, \tag{8}$$
$$G_{\text{room}} = K\, p_0 / \Delta T_{\text{app}}, \tag{9}$$
$$G_{\text{bldg}} = R\, K\, p_0 / \Delta T_{\text{app}}. \tag{10}$$

Equation (8) gives 25 W/K at every size, and Eqs. (9) and (10) give the values of Table 1.

3.2 Readings and priors

A sensor on temperature T_j reports y_j with Gaussian noise of standard deviation $\sigma_s = 0.05$ K, and each heat term at level ℓ (rack, room or building) has a zero-mean Gaussian prior with standard deviation s_ℓ:

$$y_j = T_j + \varepsilon_j, \tag{11}$$
$$\varepsilon_j \sim \mathcal{N}(0, \sigma_s^2), \tag{12}$$
$$f_\ell \sim \mathcal{N}(0, s_\ell^2). \tag{13}$$

The readings are generated from the true temperatures T_j^{*} of the fault-free building. The j-th reading of a set is perturbed by a sine of amplitude $a = 0.02$ K and phase ϕ, with $\phi = 7$ for the baseline snapshot and $\phi = 23$ for the stream, and the i-th streamed reading, counted from zero, measures rack k_i of room m_i:

$$y_j = T_j^{*} + a\sin(\phi + j), \tag{14}$$

$$k_i = (17\,i + 3) \bmod K, \tag{15}$$

$$m_i = i \bmod R. \tag{16}$$

3.3 The probabilistic model

The relations (1)–(7) are physics factors weighted with tolerance $\sigma_p = 0.001$ W, each reading (11) is a factor weighted with σ_s^{-2}, and each prior (13) a factor weighted with s_ℓ^{-2}. This subsection states how such factors are combined. A physical network is a graph, and the engine writes its physics on that graph. The state is a vector z of unknowns: one potential per node, one flow per edge, and whatever else is uncertain, such as demands or parameters. Every relation the state must satisfy is a factor, written as a residual $r_k(z)$ that vanishes when the relation holds and carrying a weight w_k, the inverse variance of how far it may be off. A conservation law at a node, a constitutive law on an edge, a sensor reading and a prior on an approximately known quantity are all factors of this one kind; they differ only in their residual and their weight, a physics relation being trusted to a small tolerance, a reading to its sensor's noise, a prior to its stated spread. Treating each factor as an independent Gaussian belief about its residual gives the posterior density of the state given the readings y,

$$p(z \mid y) \propto \prod_k \exp\!\big(-\tfrac{1}{2}\,w_k\,r_k(z)^2\big). \tag{17}$$

The state that maximizes it, the *maximum-a-posteriori* (MAP) state z^{*}, minimizes the weighted sum of squared residuals $C(z)$ and is found by Gauss–Newton iteration on the sparse Jacobian J of the residuals, $J_{ki} = \partial r_k / \partial z_i$, with W the diagonal matrix of weights, from a flat start until the gradient norm falls below a tolerance ϵ:

$$C(z) = \tfrac{1}{2}\sum_k w_k\,r_k(z)^2, \tag{18}$$

$$z^{*} = \arg\min_z C(z), \tag{19}$$

$$z \leftarrow z - (J^{\top}WJ)^{-1}J^{\top}W\,r(z) \quad \text{until } \|J^{\top}W\,r(z)\| < \epsilon. \tag{20}$$

Near the minimum the cost (18) is approximately quadratic, so the posterior is approximately Gaussian about z^{*} with information matrix Λ. This is the *Laplace approximation*, and the *posterior standard deviation* σ_i of any unknown is read from the diagonal of Λ^{-1}, from the same sparse factorization the last Gauss–Newton step used:

$$p(z \mid y) \approx \mathcal{N}\big(z^{*}, \Lambda^{-1}\big), \tag{21}$$

$$\Lambda = J^{\top}WJ, \tag{22}$$

$$\sigma_i = \sqrt{(\Lambda^{-1})_{ii}}. \tag{23}$$

When several factor graphs compete to explain the same readings — a leak here, a leak there, no leak — each is compared by its *evidence* $Z = \int \prod_k \exp(-\frac{1}{2}w_k r_k(z)^2)\,dz$, the probability of the

readings under the model once the extra freedom of its unknowns has been paid for. In the Laplace approximation the evidence has a closed form, and normalizing every factor by its own weight lets graphs with different numbers of unknowns d and factors compare as densities,

$$\log Z = -C(z^*) + \tfrac{d}{2}\log 2\pi - \tfrac{1}{2}\log\det\Lambda + \sum_k \tfrac{1}{2}\log\frac{w_k}{2\pi}. \tag{24}$$

Differences of $\log Z$ between hypotheses, from (24), are in *nats*, natural-log units of odds; five nats is odds of about 150 to 1. The posterior probability of a hypothesis is its evidence normalized over all hypotheses under equal prior odds. Everything that follows is a choice of variables and factors on this graph, followed by these three operations.

3.4 Exactness of the Gaussian posterior

Every factor of this model is linear in the unknowns: the heat balances, link laws and leakage law (1)–(7) are linear, and so are the readings (11) and the priors (13). The cost (18) is then exactly quadratic, the Gauss–Newton iteration (20) reaches its minimum, and the Laplace approximation (21) is not an approximation: the posterior is exactly Gaussian with mean $\mu = z^*$ and information matrix Λ (22). This study uses the MAP state and the posterior standard deviations; it compares no competing models, so the evidence (24) is not computed.

3.5 Sequential assimilation of one reading

Let μ and Λ be the posterior after some readings, and let a new reading y measure variable v, with e_v the unit vector of that variable and $\Sigma = \Lambda^{-1}$ the posterior covariance. Adding the reading's factor adds $\sigma_s^{-2} e_v e_v^\top$ to the information matrix. Conditioning a Gaussian on one linear observation changes its mean by a gain g times the surprise, the difference between the reading and what the current mean predicts for it, and the covariance follows from the formula of Sherman and Morrison (1950):

$$c = \Lambda^{-1} e_v, \tag{25}$$

$$g = c \,/\, (c_v + \sigma_s^2), \tag{26}$$

$$\mu^+ = \mu + g\,(y - \mu_v), \tag{27}$$

$$\Lambda^+ = \Lambda + \sigma_s^{-2}\, e_v e_v^\top, \tag{28}$$

$$\Sigma^+ = \Sigma - g\, c^\top. \tag{29}$$

The vector c of (25) is the column of the covariance belonging to the measured variable: how strongly every unknown co-varies with it. It is obtained by one pair of triangular solves against a sparse LU factorization of Λ, so the covariance is never formed. The update (27) then moves every estimate by its share of the surprise, and (28) records the reading in the information matrix. The implementation refactors Λ^+ after every reading, so that the next column comes from a fresh factorization; a rank-one modification of the factors (Davis and Hager 1999) would avoid that step, so the measured cost of the update is an upper bound on what the update requires. Because the posterior is exactly Gaussian, the mean (27) after all readings equals the MAP state of the full problem up to floating-point round-off.

3.6 Experimental protocol

At each building size the protocol of Figure 3 runs once. The *full re-solve* workflow appends each reading to the list of readings, rebuilds the model with its priors and readings and runs the MAP

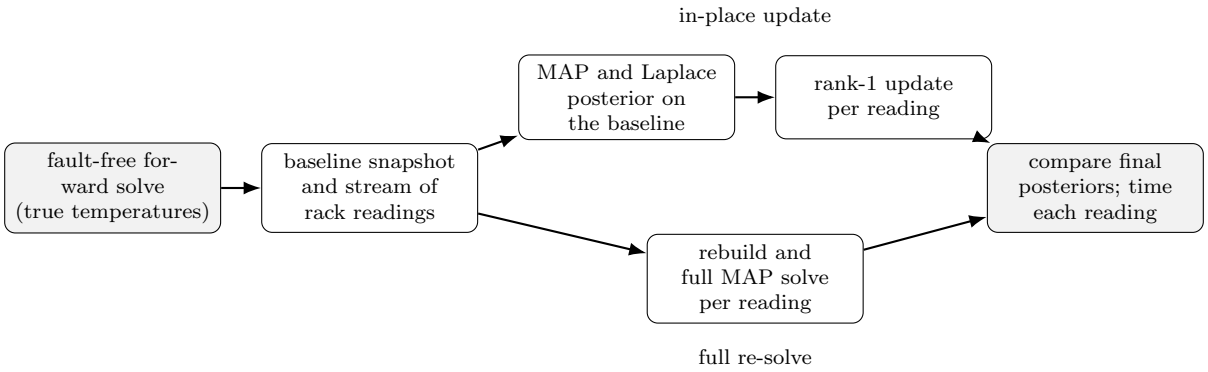


Figure 3: The protocol at one building size. A forward solve of the fault-free building supplies the true temperatures; readings are generated from them; both workflows then process the same 20 (principal case) or 5 (size sweep) streamed readings, and their posteriors after the last reading are compared. The in-place update starts from the posterior the baseline snapshot gives; the full re-solve starts from nothing at every reading.

solve (19) from the start, at most 120 iterations. The *in-place update* workflow runs the MAP solve once on the baseline snapshot and then applies (25)–(28) for each reading. After the last reading the two posterior means of every heat term are compared, and the temperatures of the forward solve and of every posterior are checked for physical consistency: heat balance at every node, rack leakage below 10 % of base power, and every temperature between 289 K and 320 K.

The per-reading time of the full re-solve runs from the rebuild of the model to the end of its MAP solve; that of the update runs from the column solve (25) to the end of the refactorization. Linear algebra is restricted to one thread. Wall-clock times depend on the machine and on its load, so the whole timing protocol is repeated five times, each repeat starting only after a check finds no other Python, Cargo or LaTeX process running and the whole machine's processor load, averaged over ten seconds, below 30 %. Every finished repeat is kept. Each timing bound is published as its minimum over the repeats, and the per-reading times of the repeat giving the smallest every-reading bound are stored.

The disagreement between the workflows is floating-point round-off, and its value depends on the order of floating-point operations, which the linear-algebra library chooses by processor. The study was therefore run under two code paths of the OpenBLAS library, those for the SkylakeX and the Haswell processor families, and every round-off quantity is reported as a cap: one decade above the smallest power of ten that bounds the larger of its two values. Every run checks its own values against the caps. Physical quantities are reported to seven significant digits and posterior standard deviations to three, at which the two code paths agree.

3.7 Metrics

Two questions are asked: do the two workflows reach the same diagnosis, and what does each cost per reading. For the first, let μ^{re} and μ^{in} be the posterior means after the stream from the full re-solve and from the update, μ^0 the mean after the baseline snapshot, and σ_j the posterior standard deviation of heat term j after the stream; j runs over all heat terms:

$$\delta = \max_j |\mu_j^{\mathrm{re}} - \mu_j^{\mathrm{in}}|, \tag{30}$$

$$\Delta = \max_j |\mu_j^{\mathrm{in}} - \mu_j^0|, \tag{31}$$

$$\rho = \max_j \left| \mu_j^{\mathrm{re}} - \mu_j^{\mathrm{in}} \right| / \sigma_j . \tag{32}$$

The disagreement δ (30), in watts, is the largest difference between the two diagnoses. Whether it is small needs a reference, and two are available. The judged criterion is $\rho \leq 10^{-6}$ (32): the difference in every heat term, measured in that term's own posterior standard deviation. The standard deviation is the uncertainty an operator reads with the estimate, so a difference far below it cannot change what the diagnosis says, and the ratio is taken term by term, so each difference is read against the uncertainty of the same term. The scales in this building make the choice consequential. The rack terms have standard deviations of 1.28 to 1.76 W, the rooms 32.0 to 555 W and the building 164 to 1,640 W over the sizes studied, while the stream moves no estimate by more than 0.48 W, and the disagreement sits in the room and building terms (Table 4). The second reference is the stream's own move Δ (31), with the criterion $\delta/\Delta \leq 10^{-6}$. It is reported, but it divides a maximum over one heat term by a maximum over another: the largest disagreement, in a room or building term with a large standard deviation, is set against the largest move, which is a fraction of a watt, and the ratio grows with the building for that reason alone. For both criteria the reference value is the tolerance, 10^{-6}.

For the cost, let t_i^{re} and t_i^{in} be the times of the two workflows for reading i in the principal case, and $\tilde{t}_K^{\mathrm{re}}$ and $\tilde{t}_K^{\mathrm{in}}$ their medians over the readings at size K. The every-reading bound B_{read} sets the fastest full re-solve against the slowest update, and the every-size bound B_{size} is the smallest ratio of the medians over the sizes:

$$B_{\mathrm{read}} = \left\lfloor \min_i t_i^{\mathrm{re}} \,/\, \max_i t_i^{\mathrm{in}} \right\rfloor , \tag{33}$$

$$B_{\mathrm{size}} = \left\lfloor \min_K \tilde{t}_K^{\mathrm{re}} \,/\, \tilde{t}_K^{\mathrm{in}} \right\rfloor . \tag{34}$$

Both are extreme statistics, chosen so that a published statement of the form "at least B times cheaper" holds for every reading or every size measured. Their reference value is one: a bound of one or less would mean no advantage.

4 Results

4.1 Physical consistency

Across the forward solves and every posterior of the study, 5,196 temperatures, the hottest is 308.07 K and the coldest 292.99 K, inside the band of 289 to 320 K. The largest rack leakage is 0.27 % of base power, against a limit of 10 %. In the forward solves heat balances at every node to a relative residual below 10^{-12}, and in the posteriors below 10^{-12}, against limits of 10^{-9} and 10^{-3}. The MAP solve converged at every size and every reading.

4.2 Agreement of the two workflows

Table 3 lists, for the principal case and every size of the sweep, the cap on the largest disagreement δ, the largest move of the stream Δ, the cap on their ratio with the outcome of the δ/Δ criterion, and the cap on ρ with the outcome of the judged criterion. The judged criterion holds at every size: in every heat term the two diagnoses differ by at most 10^{-6} of that term's posterior standard deviation. The caps on ρ rise from 10^{-8} at 16 racks to 10^{-5} at the largest sizes; a cap lies at least one decade above what either code path produced, which is why it can exceed the tolerance while every measured value lies within it. The disagreement itself grows with the building, from a cap of 10^{-6} W at 16 racks to 10^{-2} W in the principal case. The room and building conductances are

Table 3: Agreement of the full re-solve and the in-place update after the stream, principal case first and the size sweep below. δ is the largest difference of the two posterior means over all heat terms (Eq. (30)), Δ the largest move the stream made in any estimate (Eq. (31)), and ρ the largest difference measured in each term's own posterior standard deviation (Eq. (32)). Round-off quantities are given as caps (Section 3.6). The column $\delta/\Delta \le 10^{-6}$ reads *holds* or *fails* when both floating-point kernels agree and *kernel-dependent* when they do not; the last column is the judged criterion.

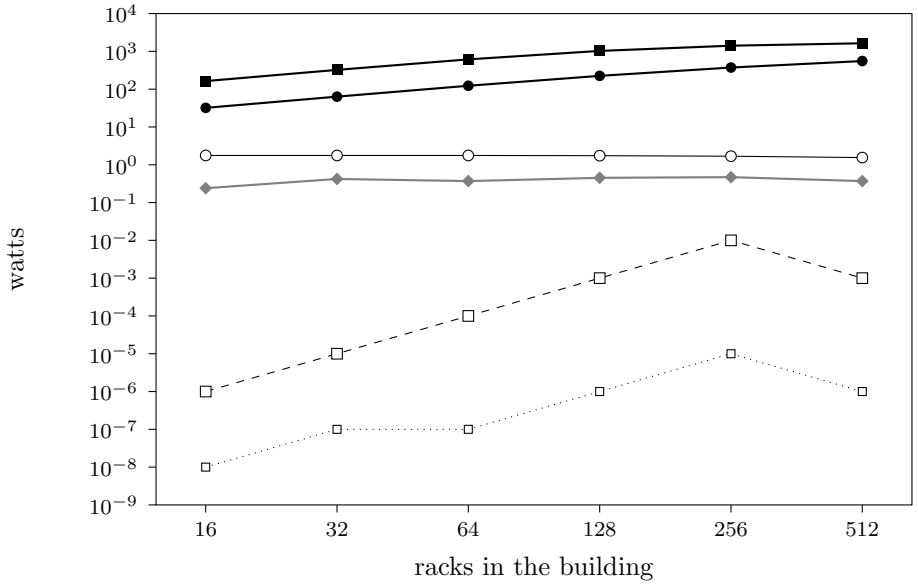
racks	unknowns	readings	δ cap [W]	Δ [W]	δ/Δ cap	$\delta/\Delta \le 10^{-6}$	ρ cap	$\rho \le 10^{-6}$
256	1,039	20	10^{-2}	0.48	10^{-2}	fails	10^{-5}	yes
16	79	5	10^{-6}	0.24	10^{-5}	holds	10^{-8}	yes
32	143	5	10^{-5}	0.42	10^{-4}	kernel-dependent	10^{-7}	yes
64	271	5	10^{-4}	0.37	10^{-4}	fails	10^{-7}	yes
128	527	5	10^{-3}	0.45	10^{-3}	fails	10^{-6}	yes
256	1,039	5	10^{-2}	0.47	10^{-2}	fails	10^{-5}	yes
512	2,063	5	10^{-3}	0.37	10^{-2}	fails	10^{-5}	yes

Table 4: Where the disagreement sits, by level of the building. The first three columns are posterior standard deviations of the heat terms after the stream (the range over racks; the four rooms agree to the digits shown); the next three are the caps on the largest disagreement δ among the rack terms, the room terms and the building term; the last is the cap on ρ. The rack terms carry uncertainties of a watt or two and disagreements capped far below them; the room and building terms carry the larger disagreements, against standard deviations of tens to thousands of watts.

racks	σ rack [W]	σ room [W]	σ building [W]	δ cap, racks [W]	δ cap, rooms [W]	δ cap, building [W]	ρ cap
256	1.43–1.68	374	1,410	10^{-5}	10^{-2}	10^{-2}	10^{-5}
16	1.44–1.76	32.0	164	10^{-8}	10^{-6}	10^{-6}	10^{-8}
32	1.53–1.76	63.4	323	10^{-7}	10^{-5}	10^{-5}	10^{-7}
64	1.52–1.76	123	613	10^{-7}	10^{-5}	10^{-4}	10^{-7}
128	1.50–1.74	225	1,030	10^{-6}	10^{-4}	10^{-3}	10^{-6}
256	1.43–1.68	374	1,410	10^{-5}	10^{-3}	10^{-2}	10^{-5}
512	1.28–1.55	555	1,640	10^{-6}	10^{-3}	10^{-3}	10^{-5}

hundreds of times larger in the large buildings than in the small ones, and the room and building heat terms are fixed by small differences between large, nearly cancelling heat flows, where round-off accumulates.

The δ/Δ criterion holds at 16 racks, fails from 64 racks upward on both code paths, and at 32 racks holds on one code path and fails on the other. Table 4 and Figure 4 show why the two criteria part. Table 4 splits the disagreement and the standard deviations by level of the building, and Figure 4 draws the standard deviations, the stream's move and the disagreement caps of the sweep on one logarithmic scale. The disagreement in the rack terms is capped at 10^{-5} W at every size, while the room and building terms carry disagreements capped up to 10^{-2} W. Those terms have standard deviations of 32.0 to 1,640 W, so measured against their own uncertainty their disagreement is of the same order as the rack terms'. The stream, in contrast, moves no estimate by more than 0.48 W, less than the standard deviation of any heat term at any size, and the δ/Δ ratio sets the room and building disagreement against that move. Its failure beyond the smallest building therefore reflects the difference of scale between the terms, not a difference between the diagnoses.



filled square: building standard deviation; filled circle: room; open circle: largest rack
diamond: largest move of the stream; dashed: cap on the largest disagreement; dotted: its cap among rack terms

Figure 4: The scales that decide which yardstick of agreement is meaningful, over the size sweep (16 to 512 racks, 5 streamed readings each; both axes logarithmic). The three solid lines with filled or open circles and squares are posterior standard deviations of the unknown heat terms after the stream: the building's (164 to 1,640 W), the rooms' and the largest among the racks' (at most 1.76 W). The gray diamonds are the largest move the stream made in any estimate, below 0.48 W at every size. The dashed line is the cap on the largest disagreement between the two workflows over all heat terms and the dotted line its cap over the rack terms alone. The disagreement grows with the building and sits in the room and building terms, whose standard deviations are tens to thousands of watts; the stream's move is smaller than every one of those standard deviations.

4.3 Cost per reading in the principal case

Figure 5 plots the time each workflow took for each of the 20 streamed readings at 256 racks, on a logarithmic axis. The full re-solve took 101 to 610 ms per reading, median 188 ms, and the in-place update 5.1 to 10.5 ms, median 8.3 ms. The two ranges do not overlap: in the repeat plotted the fastest re-solve was 9× the slowest update. Over the five repeats this bound B_{read} (33) ranged from 9 to 25, so the update was at least 9× (9–25× over five repeats) cheaper than the full re-solve on every reading, against a reference of one. The re-solve's time varies by a factor of 6.1 across the stream, the update's by a factor of 2.1.

4.4 Cost across building sizes

Figure 6 plots the median time per reading of both workflows against the number of unknowns, and Table 5 lists both timing bounds in each of the five repeats. Both medians rise with the building. The full re-solve's grows from 33.6 ms at 16 racks to 445 ms at 512 racks, and the update's from 0.45 ms to 25.5 ms: the update also grows, because its factorization and its column solve scale with the number of unknowns. In the repeat plotted the ratio of the medians ranges from 17.5 to 76.5 and is smallest at the largest building.

Over the repeats of Table 5 the every-size bound B_{size} (34) ranged from 13 to 21 and the every-

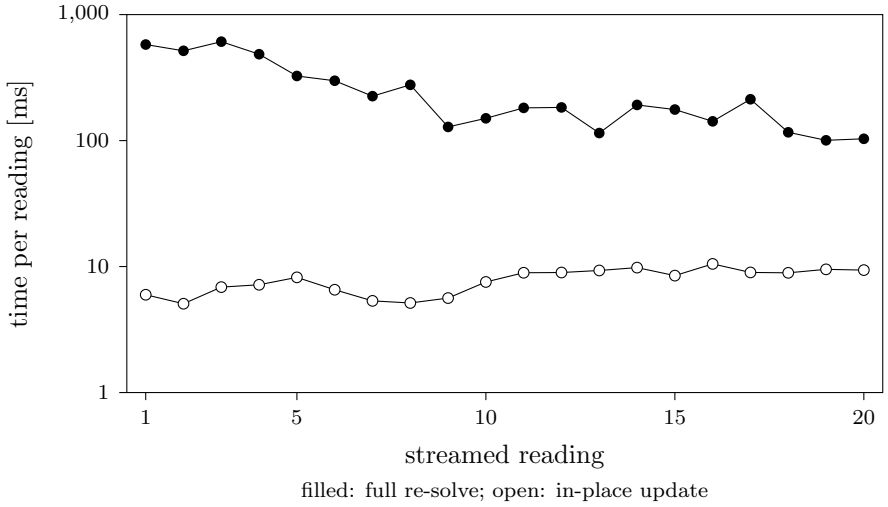


Figure 5: Wall-clock time to absorb each of the 20 streamed readings at 256 racks (1,039 unknowns), logarithmic vertical axis. Filled circles are the full re-solve (101 to 610 ms, median 188 ms), open circles the in-place update (5.1 to 10.5 ms, median 8.3 ms). The times plotted are those of repeat 2 of five, the repeat with the smallest every-reading ratio; its fastest re-solve against its slowest update gives the published bound of 9×.

reading bound from 9 to 25, with the machine's load at the start of each repeat between 7.4 and 21.4 %. The median update was at least 13× (13–21× over five repeats) cheaper than the median full re-solve at every size. The spread between repeats, a factor of 2.8 for the every-reading bound, is the reason both bounds are published as minima.

5 Discussion

The agreement found is the agreement the mathematics promises. The model is linear and its noise Gaussian, so the rank-one update and the full solve compute the same posterior, and the study measures the floating-point distance between two computations of one quantity. What it establishes is that this distance stays far below the uncertainty of every estimate on a sized network of up to 512 racks. A model with nonlinear relations, such as a leakage law that grows with temperature faster than linearly or airflow that depends on fan speed, would make the update approximate: the information matrix would have to be relinearized as the estimates move, as incremental smoothing does (Kaess et al. 2012), and the agreement would then depend on how far and how often the estimates move between relinearizations. The study does not test that case.

The margin under the tolerance is not large. The measured values of ρ lie within the tolerance of 10^{-6} at every size on both code paths, but the caps, one decade above them, reach 10^{-5}, so the largest measured values lie within about a decade of the tolerance. Round-off also decides the δ/Δ criterion at 32 racks, where the two code paths fall on opposite sides of it. A third code path, a different factorization order or a larger building could move either outcome, and every run checks its values against the published caps so that such a change is detected rather than absorbed.

The timings come from one workstation and bound rather than measure the advantage. They are extreme statistics, they varied by a factor of up to 2.8 between repeats taken minutes apart on a machine checked to be quiet, and the per-reading times of the repeat giving the smallest every-size bound were not kept, so Figure 6 shows the medians of a different repeat than the one

Table 5: The five repeats of the whole timing protocol. Each started only after a check found no other Python, Cargo or LaTeX process and a whole-machine processor load, averaged over ten seconds, below 30 %; the load that check measured is listed. The last two columns are the two bounds of Eqs. (33) and (34) in each repeat, and the published value of each is its minimum over the repeats.

repeat	started (UTC)	machine CPU [%]	every reading	every size
1	2026-09-17 16:38:38	7.4	23×	20×
2	2026-09-17 16:38:58	8.6	9×	17×
3	2026-09-17 16:40:37	21.4	16×	13×
4	2026-09-17 16:42:16	8.0	25×	21×
5	2026-09-17 16:42:35	8.5	10×	19×
published (minimum)			9×	13×

that set that bound. The update measured refactors the information matrix after every reading, so its cost grows with the building, reaching 25.5 ms at 512 racks; a rank-one modification of the factors (Davis and Hager 1999) would lower it, and the full re-solve measured rebuilds its model at every reading, which a solver that kept its model could avoid. The ratio of the two is therefore a statement about these two implementations, and the smallest ratio falls at the largest building.

The data are synthetic. The readings are the true temperatures perturbed by a deterministic sine rather than drawn noise, the building contains no fault, and the stream is short: the study measures agreement and cost, not whether the diagnosis finds a fault, which depends on sensor placement and noise and is not examined. Every relation other than the heat terms is assumed known; errors in the conductances or base powers would be absorbed into the heat-term estimates by both workflows alike.

6 Conclusion

This report asked whether a data-center building's thermal diagnosis, a Gaussian posterior over an unknown heat term at every rack, room and the building, must be solved again whenever a new temperature reading arrives, or whether the reading can be folded into the existing posterior by a rank-one update. On a building of 4 rooms whose every link is sized for its heat, at six sizes from 16 to 512 racks, the update and the full solve reached the same diagnosis: after the stream the two differ in every heat term by at most 10^{-6} of that term's posterior standard deviation. Judged instead against the largest move the stream made, at most 0.48 W while the room and building terms carry uncertainties of 32.0 to 1,640 W, the agreement holds only at the smallest building, a criterion set on the wrong scale for this network. The update was at least 9× (9–25× over five repeats) cheaper than the full solve on every reading at 256 racks, and its median at least 13× (13–21× over five repeats) cheaper at every size, on one workstation. The next steps are a rank-one modification of the factorization in place of refactoring it, and a nonlinear model in which the update must be relinearized.

A Reproduction

```
cd docs/terra_graph_engine/case_studies/level1_building/incremental_inference
PYTHONPATH=../../../../../src python run.py
PYTHONPATH=../../../../../src python make_report.py --pdf
```

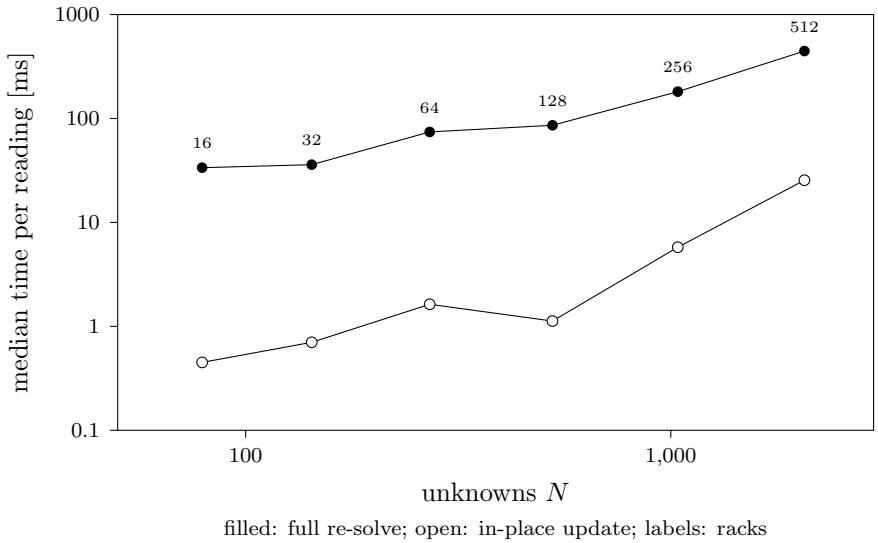


Figure 6: Median wall-clock time per streamed reading against the number of unknowns, from 16 to 512 racks (79 to 2,063 unknowns), 5 readings per size, both axes logarithmic. Filled circles are the full re-solve, open circles the in-place update. The medians plotted are those of repeat 2, whose per-reading times are stored; in that repeat the smallest ratio of the two medians is 17.5 (at 512 racks). The published every-size bound, 13×, is the smallest over the five repeats and comes from repeat 3, whose medians are not stored.

```
cd ../../../../..
python -m pytest -q tests/plugins/datacenter/test_incremental_inference_case_study.py
```

`run.py` builds the building at each size, runs both workflows on the same stream, checks every result against the caps and acceptance limits, and writes `results.json`; with `--out DIR` it writes into another directory and copies the committed `timings.json` beside the results. `make_report.py` reads only `results.json` and the `timings.json` beside it and writes `report.tex`. The readings are deterministic (phases 7 and 23, rack pattern (15)), and `results.json` is identical byte for byte under the SkylakeX and Haswell code paths of OpenBLAS. The timings are not regenerated by these commands: `run.py --retime` measures them again, five repeats after quiet-machine checks, and rewrites `timings.json` in the study directory only. The committed timings were measured on 2026-09-17 on a Windows-11-10.0.26200-SP0 workstation (Intel64 Family 6 Model 140 Stepping 1, GenuineIntel), Python 3.12.4, NumPy 2.4.6, SciPy 1.15.2, with `OPENBLAS_NUM_THREADS`=1. The test in the last line regenerates the results, compares them with the committed file and rebuilds the tex.

B Data and provenance

No measured data enter this report: the building, its loads and its readings are generated by `run.py` from the model of Section 3, and `results.json` and `timings.json` are the only inputs the report reads. Conductances follow the sizing rule of Eqs. (8)–(10). The model at commit `5be377ed` used $G = 10$ W/K (rack), 20 W/K (room) and 100 W/K (building) at every building size; its figures are not comparable. The study leaves two open problems: the same comparison on a nonlinear model, where the update must be relinearized, and the same comparison with a rank-one modification of

the sparse factorization in place of refactoring it.

C Notation

Table 6 lists the symbols of the report.

Table 6: Symbols used in the report, with units.

symbol	meaning	unit
T_r, T_m, T_b	temperature of rack r, room m, the building	K
P_r	electrical power dissipated by rack r	W
Q_r, Q_m, Q_b	heat flow from a rack to its room, a room to the building, the building to chilled water	W
f_r, f_m, f_b	unknown heat term at a rack, a room, the building	W
G_{rk}, G_{room}, G_{bldg}	thermal conductance of a link	W/K
p_0, α, T_{ref}	rack base power, leakage slope, leakage reference temperature	W, W/K, K
K, R	racks per room, rooms	—
y_j, σ_s	reading of sensor j, sensor noise	K
s_ℓ	prior standard deviation of a heat term at level ℓ	W
μ, Λ, Σ	posterior mean, information matrix, covariance ($\Sigma = \Lambda^{-1}$)	—
e_v, g	unit vector of the measured variable, gain	—
δ, Δ, ρ	disagreement, stream move, disagreement in standard deviations	W, W, —
B_{read}, B_{size}	timing bounds	—

References

[Masanet et al. 2020] Masanet, E., Shehabi, A., Lei, N., Smith, S., Koomey, J. (2020). Recalibrating global data center energy-use estimates. Science 367(6481), 984–986. doi:10.1126/science.aba3758.

[Barroso et al. 2019] Barroso, L. A., Hölzle, U., Ranganathan, P. (2019). The Datacenter as a Computer: Designing Warehouse-Scale Machines. Synthesis Lectures on Computer Architecture, Springer. doi:10.1007/978-3-031-01761-2.

[Kalman 1960] Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. Journal of Basic Engineering 82(1), 35–45. doi:10.1115/1.3662552.

[Särkkä 2013] Särkkä, S. (2013). Bayesian Filtering and Smoothing. Cambridge University Press. doi:10.1017/CBO9781139344203.

[Sherman and Morrison 1950] Sherman, J., Morrison, W. J. (1950). Adjustment of an inverse matrix corresponding to a change in one element of a given matrix. The Annals of Mathematical Statistics 21(1), 124–127. doi:10.1214/aoms/1177729893.

[Hager 1989] Hager, W. W. (1989). Updating the inverse of a matrix. SIAM Review 31(2), 221–239. doi:10.1137/1031049.

[Davis and Hager 1999] Davis, T. A., Hager, W. W. (1999). Modifying a sparse Cholesky factorization. SIAM Journal on Matrix Analysis and Applications 20(3), 606–627. doi:10.1137/S0895479897321076.

[Schweppe and Wildes 1970] Schweppe, F. C., Wildes, J. (1970). Power system static-state estimation, Part I: Exact model. IEEE Transactions on Power Apparatus and Systems PAS-89(1), 120–125. doi:10.1109/TPAS.1970.292678.

[Kschischang et al. 2001] Kschischang, F. R., Frey, B. J., Loeliger, H.-A. (2001). Factor graphs and the sum-product algorithm. IEEE Transactions on Information Theory 47(2), 498–519. doi:10.1109/18.910572.

[Kaess et al. 2008] Kaess, M., Ranganathan, A., Dellaert, F. (2008). iSAM: Incremental smoothing and mapping. IEEE Transactions on Robotics 24(6), 1365–1378. doi:10.1109/TRO.2008.2006706.

[Kaess et al. 2012] Kaess, M., Johannsson, H., Roberts, R., Ila, V., Leonard, J. J., Dellaert, F. (2012). iSAM2: Incremental smoothing and mapping using the Bayes tree. The International Journal of Robotics Research 31(2), 216–235. doi:10.1177/0278364911430419.

[Dellaert and Kaess 2017] Dellaert, F., Kaess, M. (2017). Factor graphs for robot perception. Foundations and Trends in Robotics 6(1–2), 1–139. doi:10.1561/2300000043.

[Li 2005] Li, X. S. (2005). An overview of SuperLU: Algorithms, implementation, and user interface. ACM Transactions on Mathematical Software 31(3), 302–325. doi:10.1145/1089014.1089017.

[Virtanen et al. 2020] Virtanen, P., Gommers, R., Oliphant, T. E., et al. (2020). SciPy 1.0: Fundamental algorithms for scientific computing in Python. Nature Methods 17(3), 261–272. doi:10.1038/s41592-019-0686-2.