# REDUCING NOISE IN MONTE CARLO LABELS WITH LOCAL CONSERVATION

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Machine-learning surrogates are often trained on Monte Carlo (MC) labels, whose sampling noise around the simulator's mean output is costly to reduce. A local conservation law, or balance, can help correct such a label after the run, but the label's mismatch with the balance contains both sampling noise and a persistent approximation error. We analyze a classical linear correction that uses the noise covariance and a trust level per equation to subtract the noise predicted from this mismatch. A first limit is that, when the balance is treated as exact, merging its local equations into coarser ones cannot increase the noise variance removed. A second is that, for a fixed simulator input, the label improves on average over runs exactly when this variance exceeds the squared bias from the persistent error. Since more samples shrink only the noise, one balance can improve a noisy label and degrade a cleaner one. Because this bias is unknown, a one-label diagnostic estimates a lower bound on the expected error reduction, but the estimate is noisy and does not guarantee improvement for a single label. In airport-dispersion tests at a nominal $300$k particles per run, the correction improves $91\%$ of the $34$ evaluated labels against separate reference runs, on selected cells within $0$–$3$ m of the ground. Estimating the variances and trust levels requires an extra, larger run, so the gain is in accuracy, not simulation cost. In separate experiments with more samples per run, the correction degrades many labels, as the second limit allows.

## 1 INTRODUCTION

Monte Carlo (MC) simulators estimate an average from a finite number of random samples. For one fixed set of inputs, which we call a *scene*, two runs return different outputs. A machine-learning surrogate trained on such outputs predicts new ones at low cost (Li et al., 2021a; Lu et al., 2021). An output stored for training becomes a *label*, and its sampling error becomes label noise. As samples accumulate, the label approaches the simulator's infinite-sample output, our target, which describes the simulator, not physical reality. A label's *expected squared error* is its squared distance from this target, averaged over repeated runs. For a standard MC average of independent, identically distributed samples with finite variance, doubling their number halves this error, at about twice the cost (Owen, 2013). We ask whether physics can instead improve a completed label.

Many simulated quantities obey a local balance. For a transported substance, what enters a region minus what leaves it, plus what is emitted there, minus what is removed, equals what accumulates. We consider grid balances linear in the simulator's output. The target satisfies them only approximately, because they simplify what the simulator computes. They may neglect a loss, or represent random particle motion as diffusion, an approximation called a *closure*. On a noisy label, they leave a *residual*, the difference between their two sides. Part of it is MC noise, which a correction should remove. The other part, the *closure gap*, persists as the sample count tends to infinity. Removing the whole residual would shift the corrected label's average away from the target, a systematic error, or *bias*. The problem we study is when such a balance improves a completed label.

Existing methods address parts of this problem. Variance reduction acts inside the simulator, before a label is stored (Owen, 2013). Denoisers clean an output using image structure, or two runs and a physical constraint (Bako et al., 2017; Wei et al., 2026). Data reconciliation adjusts noisy measurements to satisfy balance equations, possibly uncertain ones (Maquin et al., 2000), and physics-informed learning adds the balance to the training loss (Li et al., 2021b). Two questions matter for a



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

completed MC label. When does a scene's closure gap cost more accuracy than the noise removed? And how much of its noise can a balance reveal, depending on the scale of its equations?

Our first contribution is an analysis of a classical linear correction applied to completed MC labels. The correction subtracts from the label the MC error predicted from the residual. Its weights use the MC covariance, which gives the variance of each output value's error and how the errors of different values vary together. They also use a *design model* of the closure gap, which states how far each equation is trusted but does not estimate any scene's gap. Under this model, these fixed weights minimize the expected squared error among corrections that subtract a linear function of the residual, as in generalized least squares with uncertain constraints. For a fixed scene, the expected squared error over repeated runs splits into the MC variance left after correction and the squared bias caused by the actual closure gap. The label then improves on average exactly when the MC variance removed exceeds this squared bias. More samples shrink the MC variance but not the gap, so the same scene can gain from correction with few samples and lose with many.

Our second contribution concerns the spatial scale of the balance. One equation for the whole domain checks only a total, in which errors of opposite signs in different cells cancel. We prove that, when the correction treats the balance as exact, merging equations into coarser ones never increases the share of MC variance it removes. Finer equations can reveal more noise, but they may also approximate the simulator less well. Our third contribution is a *screen*. Given the sampling covariance and the correction weights, it uses one label to estimate a lower bound on the scene's expected gain, the fraction of expected squared error that the correction removes. This estimate is unbiased if the covariance is exact and the other inputs are fixed without using the label. A single value is noisy and can even have the wrong sign. It also applies only to the cells it sums over.

We test these predictions on a Lagrangian simulator of pollutant dispersion around airports, which follows particles carried by the wind and scattered by turbulence. The experiments estimate the sampling covariance, and some reuse the scored runs to build the correction, whereas the guarantees need the true covariance and inputs chosen without the label. In eight-run experiments with inputs estimated from those runs, all $48$ corrected runs over flat terrain, where the balance follows the simulator closely, improve against a reference from separate runs on selected near-ground cells. An idealized test, not the correction itself, projects a run's difference from a reference onto what the balance measures. One equation per cell captures nearly all of it, and one for the whole domain none. Around buildings, where the balance departs from the simulator, eight-run experiments at 3M particles per run, again with inputs from those runs, show the correction improving only $10$ of $17$ scenes for each of two pollutants. At the 300k particles of our training labels, the variances and the design model come from an independent full-budget run with ten times more particles, so a corrected label costs more simulation, not less. Under this protocol, the correction improves $31$ of $34$ labels on the scored stored-layer support. On the same support, the screen predicts the sign of the change for $31$ of these $34$, with the same sign accuracy as always predicting improvement. Label and learner gains need separate evaluations.

## 2 Related Work

**Reducing Monte Carlo noise.** Variance reduction changes the sampling process or the estimator inside the simulator (Owen, 2013). Other methods clean the output after the run. Rendering methods denoise images made from few samples (Zwicker et al., 2015; Bako et al., 2017), and smoothness-increasing accuracy-conserving (SIAC) filters smooth fields computed from the particles of plasma simulations (Picklo et al., 2024). For direct simulation Monte Carlo (DSMC) of rarefied gases, a self-supervised method combines two noisy runs with a physical constraint (Wei et al., 2026). Our focus is an explicit linear correction built from a local balance and the sampling covariance, with an analysis of when the balance's mismatch makes it harmful.

**Physical balances and noisy data.** Data reconciliation adjusts measured process variables until mass or energy balances hold (Crowe, 1996). It can account for correlated measurement errors (Hodouin et al., 1998) and weight uncertain balance equations accordingly (Maquin et al., 2000). Data assimilation combines observations with imperfect physical models (Carrassi et al., 2018). Our correction uses the linear-estimation algebra behind these methods. Stein's unbiased risk estimate (SURE) estimates an estimator's expected squared error, called its risk, from one noisy observation

(Stein, 1981). For a fixed linear residual correction, our gain estimate has the form of the error reduction that SURE estimates, divided by the expected squared error before correction. Here its unbiasedness needs only the noise covariance, not a Gaussian assumption. Our screen also subtracts the squared closure bias that the design model expects, divided by the same expected error.

**Noise and physics in operator learning.** When physics enters operator learning, it usually changes the predictor or its loss (Li et al., 2021b; Liu et al., 2024). Other work trains directly on noisy MC estimates. Samples from walk-on-spheres, an MC method for elliptic equations, supervise neural operators (Viswanath et al., 2026), and a wireless-field operator learns from labels computed with a finite number of rays (Murshed et al., 2026). Generalized least squares would weight the training loss by the inverse noise covariance (Aitken, 1935). These approaches act through the learner. We change the stored label instead, and test separately whether MC noise matters to a learner.

We use these tools to ask how the balance's scale, a scene's closure gap, the sample count and the scored cells decide whether correcting a completed MC label helps, and whether one label can tell.

## 3 THEORY OF LOCAL-CONSERVATION LABEL CORRECTION

We answer these questions for any MC simulator that provides a noisy label, the covariance of its noise, and linear balance equations that its infinite-sample output approximately satisfies.

### 3.1 FROM A NOISY LABEL TO A CORRECTION

Consider one *scene*, a fixed set of simulator inputs. A run with $N$ MC samples returns a label $\tilde{c} \in \mathbb{R}^{n_c}$, a vector of $n_c$ real numbers such as the values of a field on $n_c$ grid cells. Because the samples are random, each run returns a different label. We write $\mathbb{E}_\varepsilon$ for the expectation, the average over infinitely many runs. The target of the label is $c^\star := \mathbb{E}_\varepsilon \tilde{c}$, which need not be physical truth. Its error $\varepsilon := \tilde{c} - c^\star$ has zero mean by construction. The covariance $\Sigma_N := \mathbb{E}_\varepsilon[\varepsilon\varepsilon^\top]$, where $^\top$ is the transpose, holds the variances on its diagonal and, off it, how pairs of errors vary together.

**Assumption 1** (Known sampling covariance)**.** The covariance $\Sigma_N$ is known and positive definite, written $\Sigma_N \succ 0$, that is, $v^\top \Sigma_N v > 0$ for every vector $v \neq 0$.

For standard MC, the label averages $N$ independent, identically distributed contributions. Then $c^\star$ does not depend on $N$, and $\Sigma_N = \Sigma_1/N$ with $\Sigma_1$ the covariance of one contribution. The expected squared error $\mathbb{E}_\varepsilon\|\varepsilon\|^2$, with $\|x\|^2 := x^\top x = \sum_i x_i^2$, is the sum of the variances, the trace $\operatorname{tr}\Sigma_N$ (the sum of its diagonal). It vanishes as $N$ grows, making $c^\star$ the infinite-sample output.

A *balance* is a set of $n_r$ linear equations $Lc = q$ that $c^\star$ should satisfy, with a matrix $L \in \mathbb{R}^{n_r \times n_c}$ and a vector $q \in \mathbb{R}^{n_r}$ chosen without using the label, so that they stay fixed when we average over runs. Each row of $L$ combines the values of neighbouring cells into one local quantity, such as the net outflow of a conserved substance from a cell. The matching entry of $q$ is the value it should take, such as the emission in the cell. $L$ rests on a discretization and on physical approximations, such as a *closure*, which expresses unresolved motions like turbulence through the resolved field. Hence $c^\star$ misses the balance by a fixed *closure gap* $\bar{\delta} := Lc^\star - q$, the same in every run (the bar marks a fixed vector, not an average). The *residual* is

$$r := L\tilde{c} - q = (Lc^\star - q) + L\varepsilon = \underbrace{\bar{\delta}}_{\text{closure gap}} + \underbrace{L\varepsilon}_{\text{MC noise}} . \tag{1}$$

The residual alone cannot separate the two. Yet across runs, $\bar{\delta}$ stays fixed while $L\varepsilon$ averages out.

We correct the label linearly from its residual, $\hat{c}_K := \tilde{c} - Kr$. The *correction matrix* $K \in \mathbb{R}^{n_c \times n_r}$, chosen without using the label, sets how much the residual of each equation changes each entry. By (1), the corrected label has error $\hat{c}_K - c^\star = \varepsilon - K(\bar{\delta} + L\varepsilon) = (I - KL)\varepsilon - K\bar{\delta}$, where $I$ is the identity matrix. This error has a random part and a *bias* $-K\bar{\delta}$, repeated identically in every run. Since $\varepsilon$ has zero mean and a noise $A\varepsilon$ has expected squared length $\mathbb{E}_\varepsilon \operatorname{tr}(A\varepsilon\varepsilon^\top A^\top) = \operatorname{tr}(A\Sigma_N A^\top)$,

$$\mathbb{E}_\varepsilon\|\hat{c}_K - c^\star\|^2 = \underbrace{\operatorname{tr}\left[(I - KL)\Sigma_N(I - KL)^\top\right]}_{\text{MC variance left}} + \underbrace{\|K\bar{\delta}\|^2}_{\text{squared bias}} . \tag{2}$$

Choosing $K$ therefore trades the noise left in the label against the bias written into it.

The best $K$ depends on $\bar{\delta}$, which is unknown, and we only roughly know its size in each equation. A *design model* encodes this knowledge to choose $K$. It treats the gap as a random vector $\delta$. Its *second moment* $\Sigma_\delta := \mathbb{E}_\delta[\delta\delta^\top]$ includes both the spread of the modelled gap and any nonzero mean.

**Assumption 2** (Closure-gap design model)**.** The design model draws $\delta \in \mathbb{R}^{n_r}$ uncorrelated with $\varepsilon$, $\mathbb{E}_{\varepsilon,\delta}[\varepsilon\delta^\top] = 0$, where $\mathbb{E}_{\varepsilon,\delta}$ averages over both, and $\Sigma_\delta$ is chosen without using the label.

$\Sigma_\delta$ encodes how far each equation is trusted, since a large diagonal entry marks an equation expected to be violated. Under the design model, $\delta$ replaces $\bar{\delta}$ in (1), so $r = \delta + L\varepsilon$. With one entry and one equation ($L = 1$, $\Sigma_N = \sigma^2$, $\Sigma_\delta = \sigma_\delta^2$), we estimate $\varepsilon$ by $kr$. Choosing $k$ to minimize $\mathbb{E}_{\varepsilon,\delta}[(\varepsilon - kr)^2]$ gives $k = \mathbb{E}_{\varepsilon,\delta}[\varepsilon r]/\mathbb{E}_{\varepsilon,\delta}[r^2] = \sigma^2/(\sigma^2 + \sigma_\delta^2)$. With $\hat{c} = \tilde{c} - kr$, a fully trusted equation ($\sigma_\delta = 0$) removes the whole residual, and an unreliable one ($\sigma_\delta \gg \sigma$) leaves the label almost unchanged. In general, $r$ has second moment $\Sigma_r := \mathbb{E}_{\varepsilon,\delta}[rr^\top] = \Sigma_\delta + L\Sigma_N L^\top$, and the error $(I - KL)\varepsilon - K\delta$ has

$$\Gamma(K) := \mathbb{E}_{\varepsilon,\delta}\big[(\hat{c}_K - c^\star)(\hat{c}_K - c^\star)^\top\big] = (I - KL)\Sigma_N(I - KL)^\top + K\Sigma_\delta K^\top, \tag{3}$$

whose trace is the expected squared error under the design model.

**Theorem 1** (Optimal linear residual correction)**.** *Under Assumptions 1–2, suppose $\Sigma_r \succ 0$, as holds when $\Sigma_\delta \succ 0$, and let $K^\star := \Sigma_N L^\top \Sigma_r^{-1}$ and $\Sigma_{\mathrm{post}} := \Sigma_N - K^\star L\Sigma_N$. For every fixed $K$,*

$$\Gamma(K) = \Sigma_{\mathrm{post}} + (K - K^\star)\Sigma_r(K - K^\star)^\top. \tag{4}$$

*The trace of the last term is positive unless $K = K^\star$. Hence $K^\star$ is the unique minimizer of the expected squared error among corrections $\tilde{c} - Kr$, $\Gamma(K^\star) = \Sigma_{\mathrm{post}}$, and the corrected label is*

$$\hat{c} := \tilde{c} - K^\star r = \tilde{c} - \Sigma_N L^\top\big(L\Sigma_N L^\top + \Sigma_\delta\big)^{-1}(L\tilde{c} - q). \tag{5}$$

Since $\mathbb{E}_{\varepsilon,\delta}[\varepsilon r^\top] = \Sigma_N L^\top$, $K^\star = \mathbb{E}_{\varepsilon,\delta}[\varepsilon r^\top]\,\mathbb{E}_{\varepsilon,\delta}[rr^\top]^{-1}$ is the matrix form of $k$, and (5) subtracts the error predicted from $r$. $\Sigma_{\mathrm{post}}$ is then the second moment of the error left under the design model, closure part included. This is the classical generalized-least-squares update with uncertain constraints (Aitken, 1935; Maquin et al., 2000). Appendix B proves all results of this section. With $\Sigma_N$ only positive semidefinite, $\Sigma_N \succeq 0$, so that some directions have zero variance, Theorem 1, Proposition 2 and Corollary 1 still hold if $\Sigma_r \succ 0$ and the gains below have a positive denominator, while Proposition 1(ii)–(iv) keeps $\Sigma_N \succ 0$ (Appendix B.1).

### 3.2 What the balance can reveal

The correction sees the MC error only through $L\varepsilon$. Consider two entries whose errors have equal size and opposite signs. An equation that measures only their sum sees zero, although both entries are wrong. More generally, the changes $v$ of the label with $Lv = 0$ are invisible to the balance and form its *null space*. Every change of the label splits into a part in the null space and a part that the balance measures, a sum of multiples of the rows of $L$. Such sums form the *row space* $\mathcal{R}(L)$. One global equation, such as total mass, measures a single direction, along its row. By contrast, $n_c$ linearly independent equations measure every direction. Fully trusted, they would replace the label by the balance's solution $L^{-1}q$. A nonzero $\Sigma_\delta$ keeps the simulator where the balance is doubtful. We measure the benefit of the correction at three levels, namely what the design model predicts, what one scene gains on average over runs, and what one label lets us estimate. Each is a fraction of the expected squared error before correction, $D := \operatorname{tr}\Sigma_N > 0$. A gain of $0.3$ thus means $30\%$ less expected squared error, and $-0.3$ means $30\%$ more.

**Definition 1** (Design-predicted gain)**.** $\phi_{\mathrm{design}} := (D - \operatorname{tr}\Sigma_{\mathrm{post}})/D$ is the fraction of $D$ removed under the design model, since $\operatorname{tr}\Sigma_{\mathrm{post}} = \operatorname{tr}\Gamma(K^\star)$ is the expected squared error after correction.

**Proposition 1** (Local versus global balances)**.** *In the setting of Theorem 1, (i) $\phi_{\mathrm{design}} \in [0, 1]$. Suppose now that the design treats the balance as exact, $\Sigma_\delta = 0$, and that the equations of each balance are linearly independent, so that $\Sigma_r \succ 0$. Then (ii) two balances with the same row space give the same $\phi_{\mathrm{design}}$; (iii) if $L_{\mathrm{coarse}} = AL_{\mathrm{fine}}$ for a matrix $A$, so that each coarse equation sums multiples of fine ones, then $\phi_{\mathrm{design}}(L_{\mathrm{coarse}}) \le \phi_{\mathrm{design}}(L_{\mathrm{fine}})$; and (iv) if all entries have uncorrelated errors of equal variance, a single equation gives $\phi_{\mathrm{design}} = 1/n_c$.*

When $\Sigma_\delta = 0$, $\Gamma(K^\star)$ is the covariance of the noise left, and $\phi_{\mathrm{design}}$ the fraction of MC variance removed. Merging equations can then only hide noise, and by (iv) one equation over $10^6$ such cells removes $10^{-6}$ of it. An inexact finer balance may see more noise yet do worse if less faithful.

### 3.3 When an inexact balance makes the correction harmful

We now leave the design model and return to one scene with its fixed gap $\bar{\delta}$. The corrected label then carries the bias $-K^\star\bar{\delta}$, and we call $b := \|K^\star\bar{\delta}\|^2$ the *squared closure bias*. The design model expected a random bias $-K^\star\delta$, of mean squared length $\tau_\delta := \mathbb{E}_\delta\|K^\star\delta\|^2 = \operatorname{tr}(K^\star\Sigma_\delta K^{\star\top})$.

**Proposition 2** (Closure bias)**.** *Let Assumption 1 hold, with $\Sigma_\delta$ chosen without using the label, $K^\star$ as in Theorem 1 and $\Sigma_r \succ 0$. For one fixed scene, the fraction of expected squared error that the correction removes, $\phi_{\text{scene}} := (D - \mathbb{E}_\varepsilon\|\hat{c} - c^\star\|^2)/D$, is*

$$\phi_{\text{scene}} = \phi_{\text{design}} - \frac{b - \tau_\delta}{D}. \tag{6}$$

*The correction thus improves this scene's label on average over runs if and only if $b - \tau_\delta < D\phi_{\text{design}}$.*

With $K = K^\star$, the trace of (3) gives $\operatorname{tr}\Sigma_{\text{post}} - \tau_\delta = D(1 - \phi_{\text{design}}) - \tau_\delta$ for the MC variance left in (2). Adding $b$ gives $\mathbb{E}_\varepsilon\|\hat{c} - c^\star\|^2$. Subtracting it from $D$ and dividing by $D$ gives (6). For a gap drawn from the design model, $b$ averages to $\tau_\delta$ and $\phi_{\text{scene}}$ to $\phi_{\text{design}}$. Sample size shifts this trade-off. For standard MC, $D = \operatorname{tr}\Sigma_1/N$ shrinks as $N$ grows, whereas $\bar{\delta}$ does not. If the design trusts the balance fully, $\Sigma_\delta = 0$ and $\tau_\delta = 0$, and the factors $1/N$ cancel in $K^\star = (\Sigma_1/N)L^\top(L\Sigma_1 L^\top/N)^{-1} = \Sigma_1 L^\top(L\Sigma_1 L^\top)^{-1}$. Then $K^\star$, $b$ and $\phi_{\text{design}}$ do not depend on $N$, but $b/D = Nb/\operatorname{tr}\Sigma_1$ grows like $N$. Since $LK^\star = I$, $b > 0$ on the whole field whenever $\bar{\delta} \neq 0$. Whenever $b > 0$ on the scored entries, the correction harms once $N > \phi_{\text{design}}\operatorname{tr}\Sigma_1/b$. A balance can thus help noisy labels and harm cleaner ones.

### 3.4 Screening from one label

Proposition 2 involves the unknown gap. We therefore estimate $b$ from the label, giving a diagnostic that we call a *screen*. The applied correction $\tilde{c} - \hat{c} = K^\star r = K^\star\bar{\delta} + K^\star L\varepsilon$ mixes the bias term with a noise of known expected squared length $\tau := \operatorname{tr}(K^\star L\Sigma_N L^\top K^{\star\top})$. As in (2), $\mathbb{E}_\varepsilon\|\tilde{c} - \hat{c}\|^2 = b + \tau$. The statistic $\widehat{b} := \|\tilde{c} - \hat{c}\|^2 - \tau$ is therefore an *unbiased* estimate of $b$, that is, it averages to $b$ over runs. A single $\widehat{b}$ can be negative. We also estimate a stricter gain, $\phi_{\text{net}} := \phi_{\text{design}} - b/D$, which gives no credit for the expected bias $\tau_\delta$ and equals $\phi_{\text{scene}}$ when $\tau_\delta = 0$.

**Corollary 1** (One-label gain estimates)**.** *Under the assumptions of Proposition 2, $\widehat{\phi}_{\text{scene}} := \phi_{\text{design}} - (\widehat{b} - \tau_\delta)/D$ and $\widehat{\phi}_{\text{net}} := \phi_{\text{design}} - \widehat{b}/D$ are unbiased estimates of $\phi_{\text{scene}}$ and of $\phi_{\text{net}}$.*

Since $\Sigma_r K^{\star\top} = L\Sigma_N$, $\tau + \tau_\delta = \operatorname{tr}(K^\star\Sigma_r K^{\star\top}) = \operatorname{tr}(K^\star L\Sigma_N) = D\phi_{\text{design}}$, hence $\widehat{\phi}_{\text{scene}} = \phi_{\text{design}} + (D\phi_{\text{design}} - \|\tilde{c} - \hat{c}\|^2)/D$. The screen accepts when $\widehat{\phi}_{\text{net}} \geq 0$, equivalently when $\|\tilde{c} - \hat{c}\|^2 \leq D\phi_{\text{design}} + \tau$. A larger correction raises the estimated bias, which is noisy. Since $\tau_\delta \geq 0$, the test $\widehat{\phi}_{\text{net}} \geq 0$ is stricter than $\widehat{\phi}_{\text{scene}} \geq 0$. Both estimates target an average over runs, not the gain of one label, so a single estimate can have the wrong sign (Appendix B.5). They also require the true $\Sigma_N$. Keeping only positive estimates would bias them upward, since it keeps the lucky draws. Every sum of squares and trace can be restricted to entries chosen without using the label, such as those stored for training. $D$ then becomes the sum $D_M > 0$ of their variances, and Sections 3.3–3.4 and Proposition 1(i)–(iii) still hold.

## 4 Applying the correction to a dispersion simulator

We now apply this theory to a GPU reimplementation of AUSTAL 3.3 (Ingenieurbüro Janicke, 2024), a *Lagrangian* simulator of atmospheric dispersion that follows individual particles. Each particle, an MC sample, carries an equal share of the emitted pollutant. A scene fixes the emission sources, the buildings and the weather. The wind carries the particles away from the sources, and turbulence, the air's irregular eddies, moves them at random. The concentration in a cell of a 3-D grid is the particle mass it holds, averaged over time, divided by its volume. This whole field is the label $\tilde{c}$ of Section 3. The balance couples neighbouring layers, so we correct the whole field and store its lowest layer, 0–3 m above ground, as the training label.

Besides the emissions $q$, the correction needs $L$, $\Sigma_N$ and $\Sigma_\delta$. Each scene is run twice, independently. The *label run*, with nominal budget $N = 300$k particles, gives the field $\tilde{c}$ that we correct, at the

budget of our training labels. The *full-budget run*, with nominal budget $N_{\text{full}} = 3\text{M}$ particles, gives a less noisy field $c^{\text{full}}$ and an error estimate for each cell. Since this run is separate, its information does not depend on the label's noise, as Section 3 requires, with one exception given below. A corrected label therefore costs a nominal 3.3M particles, more than the full-budget field alone. This study tests whether the correction improves labels at the training budget, and obtaining its inputs without a full-budget run is outside its scope.

### 4.1 Building the three quantities

**The balance $L$.** $L$ writes the steady advection–diffusion equation $\nabla \cdot (u\,c - \kappa \nabla c) = q$ in finite differences, with one equation per cell outside buildings. Derivatives become differences between neighbouring cells. Here $u\,c$ is transport by the wind $u$, and $-\kappa \nabla c$ is turbulent mixing represented as diffusion. The diffusivity $\kappa$ is a $3 \times 3$ matrix, since mixing differs along the wind, across it and vertically. The left side is the net outflow per unit volume and time, what leaves a cell minus what enters it, and the right side the emission per unit volume and time, which gives $q$. Treating the field as steady, with nothing accumulating in a cell, is an approximation, reasonable because the weather is fixed and the average starts after a spin-up, an initial period left out while start-up effects fade. The wind and $\kappa$ are computed before any particle moves, so $L$ carries no MC noise. The simulator moves particles, whereas $L$ treats their mixing as diffusion on the grid. This replacement is a turbulence closure. The simulator also orients the mixing along the wind the scene would have without buildings. $L$ instead follows the local wind, so they disagree near buildings. $L$ also ignores deposition, the loss of particles to the ground. This is exact for nitrogen oxides ($\text{NO}_x$), which the simulator never deposits, but not for PM-1, the finest class of particulate matter in AUSTAL, `pm-1`. These differences and the finite grid form the closure gap $\bar{\delta}$.

**The MC covariance $\Sigma_N$.** The correction receives an estimate of $\Sigma_N$, for which we keep the same symbol. We write $\Sigma_N = \text{diag}(\sigma)\, R\, \text{diag}(\sigma)$, where $\sigma_i$ is the standard deviation of cell $i$'s error and $R$ holds the correlations between cells, their covariances divided by the product of the two standard deviations. The simulator splits each run's particles into ten independent groups. From the spread of the group results, it reports for each cell a relative error $s_i$, its standard deviation over its value. In the full-budget run, cell $i$ thus has a standard deviation of about $s_i\, c_i^{\text{full}}$. The label run has ten times fewer particles, so by the $1/N$ law we estimate its variance as ten times larger, $\sigma_i^2 = (N_{\text{full}}/N)\,(s_i\, c_i^{\text{full}})^2$. A cell that no particle of the full-budget run reaches gets $\sigma_i = 0$ and stays unchanged, since its row of $K^\star$ is zero. $\Sigma_N$ is then only positive semidefinite, a case that Section 3.1 covers. Errors are correlated because a particle crosses several neighbouring cells, mostly along the wind. We model $R$ by a correlation decaying with distance and stretched along the wind. We fit it layer by layer to the between-cell correlations measured across the particle groups of the label run. We neglect error covariances between layers, as adjacent layers correlate by only $0.02$ to $0.05$. For $\text{NO}_x$, $R$ thus comes from the label's own run, which departs from the independence of Section 3. Section 5 reports controls. The eight-run experiments use different correlation fits, with shared seeds on flat scenes and a separate run on airport scenes (Appendix A). PM-1 reuses the $\text{NO}_x$ correlations of the scene, from a run independent of its label, assuming deposition leaves them intact.

**Trust in the balance, $\Sigma_\delta$.** Corollary 1 asks that $\Sigma_\delta$, the trust in each equation, be chosen without using the label. Built from the label's own residual, it could grow where that residual is large and weaken the correction exactly there. We build it from the full-budget run instead, which has the same closure gap as the label but a tenth of its MC variance. We take $\Sigma_\delta$ diagonal and square the net outflow $(Lc^{\text{full}})_i$ of that run, not its residual $(Lc^{\text{full}} - q)_i$, and subtract its approximate MC variance,

$$(\Sigma_\delta)_{ii} = \max\Big(0,\ (Lc^{\text{full}})_i^2 - \tfrac{N}{N_{\text{full}}}\big[L\,\text{diag}(\sigma^2)\,L^\top\big]_{ii}\Big) + \gamma. \tag{7}$$

The average of a square is the squared average plus the variance, so the entry approximates $(q_i + \bar{\delta}_i)^2$. Away from sources, $q_i = 0$ and it estimates the squared gap, so an equation that the full-budget run violates is trusted less. Where the emission dominates, the entry is close to $q_i^2$, so source cells are trusted little. This is a heuristic precaution, since $q$ spreads each emission evenly over its cell whereas the simulator releases it at a point. With cell errors treated as independent, the subtracted variance is $1.05$ to $1.64$ times the variance computed with the correlated covariance model in the six deployment scenes measured. The maximum with zero keeps entries nonnegative, and the small $\gamma > 0$ leaves no equation fully trusted and makes $\Sigma_r$ invertible. $\Sigma_\delta$ is thus a trust model, the design

second moment of Assumption 2, not an estimate of the gap. These choices do not bias the screen when $\Sigma_N$ is the true covariance and all correction inputs are fixed independently of the label.

### 4.2 Correcting and screening a label

The corrected label is then (5), with $K^\star r = \Sigma_N L^\top y$ where $\Sigma_r y = r$. We find $y$ by conjugate gradients, an iterative method that never stores $\Sigma_r$ and only needs products $\Sigma_r v = L(\Sigma_N(L^\top v)) + \Sigma_\delta v$. The traces in $\phi_{\mathrm{design}}$ and $\tau$ also involve matrices we never form. We estimate them from 24 random vectors, 12 on flat scenes (Hutchinson, 1990), without bias, to a relative standard error below 2% where checked. We score the correction on a *support*, a set of cells chosen from the full-budget field as Section 3.4 allows. A support keeps the cells whose full-budget concentration exceeds $10^{-3}$ of its maximum, away from walls and lateral domain edges, with a source-region exclusion on flat scenes. We use two supports. The *stored-layer support* is the filtered subset of the 0–3 m layer stored for training. The *field above* $9\,m$ is 3-D and leaves out the three lowest layers, keeping one-sided differences at the ground out of its equations. It tests the correction away from the ground. The screen estimates the gain on the cells it sums over. For stored-label evaluation, we restrict its sums and traces to the scored support within the stored layer, and Section 5 reports it above 9 m too. A label passes the screen if $\widehat{\phi}_{\mathrm{net}} \geq 0$, that is, if $\widehat{b}$ does not exceed the predicted reduction $D\phi_{\mathrm{design}}$. We fixed this rule before comparing the screen with any measured gain, and Appendix C.4 reports both estimates. Here the screen only predicts benefit. It selects nothing, and every label is kept. The full-budget run cannot independently score a label it helped correct. Section 5 therefore uses extra runs. Flat scenes, with one source and no buildings, and airport scenes are run eight times each, which gives $\Sigma_\delta$. A flat run is scored against a 12M-particle reference, an airport run against the mean of the other seven. Section 5 also scores 300k labels corrected as deployed.

## 5 Experiments

Section 3 predicts two limits. The balance must be local enough to see the noise, and faithful enough that its closure gap costs less than the noise it removes. Flat scenes test a faithful balance and the role of spatial scale, and airport scenes with buildings the closure gap. We then ask whether the noise left in a label matters to a learner. Scenes fall in six atmospheric stability classes, from very stable (I), with weak turbulent mixing, to convective (V), with strong mixing. As $c^\star$ is unobservable, we score a correction against a reference $c_{\mathrm{ref}}$ sharing no particles with the label, by its *reference contrast*

$$\phi_{\mathrm{meas}} = \frac{\|\tilde{c} - c_{\mathrm{ref}}\|_M^2 - \|\hat{c} - c_{\mathrm{ref}}\|_M^2}{\|\tilde{c} - c_{\mathrm{ref}}\|_M^2 - v_{\mathrm{ref}}}. \tag{8}$$

Here $\|x\|_M^2$ sums the squared values of $x$ over a support of Section 4.2, by default the stored layer. The denominator subtracts the reference's MC variance $v_{\mathrm{ref}}$ to estimate on average the label's squared error, and a run counts only if this denominator is positive. A positive contrast means the corrected label is closer to the reference, a negative one farther. It approximates the fraction of error removed, net of added bias, but reference noise and reuse affect it. Our multi-run experiments take correction inputs from the scored runs, and the references also shape the support and the correction, on flat scenes only through thresholds. Our controls reduce or remove this reuse. Scenes report medians over runs.

### 5.1 When the balance is faithful

On flat scenes, with one non-depositing source, $L$ follows the simulator closely. For each class we run one flat scene eight times with 250k particles. We correct each run with $\Sigma_N$ and $\Sigma_\delta$ estimated from these runs and from runs at larger budgets, and score it against a 12M-particle reference from separate runs. On the scored support within the stored layer, the median contrast over the eight runs is 0.68 to 0.80 across the six classes, and all 48 runs improve (Figure 1a). The predicted $\phi_{\mathrm{design}}$, 0.82 to 0.86, exceeds this contrast by 0.04 to 0.09, and by 0.15 in the most stable class, where neighbouring errors correlate most. With the scored run left out of $\Sigma_N$, though not of $\Sigma_\delta$, the median contrast stays between 0.62 and 0.72. With $c$ the label before or after correction, the relative error $\|c - c_{\mathrm{ref}}\|_M / \|c_{\mathrm{ref}}\|_M$ falls from 5–6% to about 3%. Generic smoothers do worse, even though their parameters are tuned to minimize error against the reference. A Gaussian filter leaves 3.2 to 4.0%, and total-variation denoising, an edge-preserving smoother, 3.9 to 4.2%.

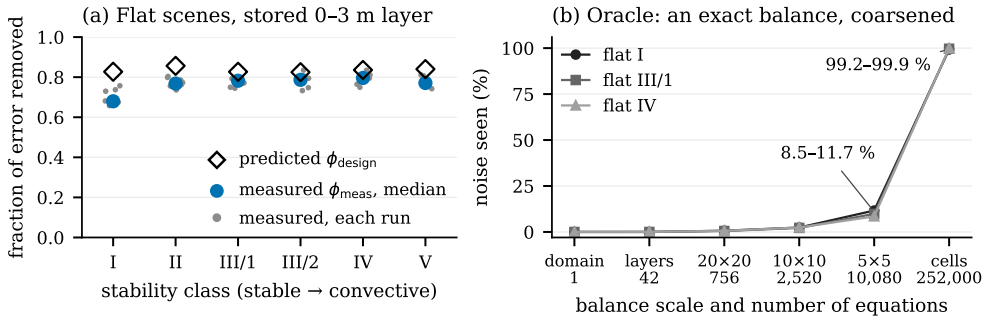


Figure 1: When the balance is faithful. (a) Flat scenes, scored support within the stored 0–3 m layer. Predicted gain, and reference contrast of each 250k run (grey) and their median (blue). (b) Oracle, not the correction. Share of the run-minus-reference difference removed by projection onto an exact balance, as its cell equations are merged (three flat classes, three 2M runs each, field above 9 m).

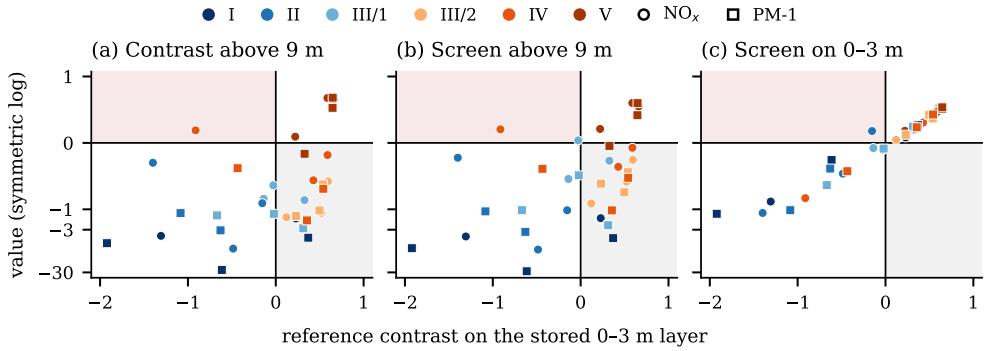


Figure 2: When the balance is not faithful. Each point is a scene with buildings (colour: stability class; marker: pollutant), summarized over its eight 3M runs and placed by its reference contrast on the scored support within the stored 0–3 m layer. (a) Contrast of the same corrections above 9 m. (b) Screen $\widehat{\phi}_{\text{net}}$ above 9 m. (c) Screen on the stored-layer support. Red: positive, yet the stored-layer contrast is negative; grey: the reverse. The vertical axes are linear between $-1$ and 1 and logarithmic in magnitude beyond.

**Locality.** An *oracle* experiment, which uses the reference field itself, measures how much noise coarser balances hide (Proposition 1). For three flat classes, we take each 2M-particle run minus the reference, mostly the run's noise. We project it onto the row space $\mathcal{R}(L)$ of a balance that the reference satisfies exactly, and measure how much removing this projection reduces its squared norm above 9 m (Figure 1b). One equation for the whole domain removes none of it, one per layer at most $0.02\%$, and one per block of $5 \times 5$ cells $8.5$ to $11.7\%$. Only cell equations remove nearly all of it, over $99\%$. This oracle shows what a balance can expose, not our correction's gain.

### 5.2 When the balance is not faithful

Near buildings, $L$ departs from the simulator. We run 18 airport scenes, three per class, eight times each with nominal budgets of 3M particles, for $NO_x$ and PM-1. We correct each run with $\Sigma_\delta$ and the variances of $\Sigma_N$ estimated from the eight runs, and score it against the mean of the other seven. In one scene per pollutant no stored-layer cell clears the concentration threshold, leaving 17 scored scenes. The correction improves 10 of them for each pollutant, and the outcome depends on the stability class (Figure 2a). The class medians are negative in the three most stable classes, down to $-0.54$ for $NO_x$ and $-0.86$ for PM-1, and positive in the three others, up to $+0.59$ and $+0.65$. In a control without reuse, where three runs build the correction and the five others are scored, the class medians keep their signs for both pollutants, with mostly lower values and different scoring supports (Appendix C.3). With ten times fewer particles, as in our training labels, and correction inputs estimated as in Section 4, the correction improves 17 of 17 $NO_x$ and 14 of 17 PM-1 labels

on the scored support within the stored layer. This is consistent with the same closure gap weighing less against more noise, although the correction inputs also differ between protocols. Above 9 m, a *plug-in control* subtracts the eight runs' mean residual to approximate removal of the gap, yielding a positive *noise-only gain* for 33 of 36 scene–pollutant pairs. Yet with the original residuals, 30 of 36 total reference contrasts are negative, with class medians down to $-4.18$ for $\mathrm{NO}_x$ and $-6.21$ for PM-1. In most scenes the closure bias thus appears to outweigh the noise removed (Proposition 2).

**Screening.** Scene by scene, the stored-layer screen $\widehat{\phi}_{\mathrm{net}}$, averaged over runs, has the sign of the scene's contrast for 16 of 17 $\mathrm{NO}_x$ and all 17 PM-1 scenes (Figure 2c). As a decision, although no label is discarded, it would accept one harmful $\mathrm{NO}_x$ correction and reject none that helps. Always predicting improvement would be right for only 20 of these 34 scene–pollutant pairs. Computed above 9 m, it matches the contrast there for 17 of 18 and all 18 scenes, but the stored label's for only 8 and 9 of 17 (Figure 2b). For one 300k label corrected as in Section 4, the stored-layer screen is right for 16 of 17 $\mathrm{NO}_x$ and 15 of 17 PM-1 labels ($31/34$, also achieved by always predicting improvement), but would accept two harmful PM-1 corrections and reject one beneficial $\mathrm{NO}_x$ correction. The predefined sign criterion is met for $\mathrm{NO}_x$ but not PM-1 (Appendix C.4).

### 5.3 Does a cleaner label help the learner?

We next test whether less label noise improves our learner. We vary the MC budget of the $\mathrm{NO}_x$ training labels, 300k or 3M particles, without applying the correction, each 300k label using a tenth of its 3M run's particles. A factorized Fourier neural operator, mapping a scene's input fields to its 0–3 m layer, is trained with each budget on the same $2{,}000$ scenes and configuration, with three random seeds. On each validation scene we subtract the two models' squared errors against the same 3M label, 300k minus 3M. The median of these differences over scenes, then over the three seeds, does not meet our prespecified heuristic criterion, three times the seed-to-seed spread of mean squared error, and the difference in mean squared error changes sign across seeds. A separate diagnostic puts another operator's typical validation error at about 31 times the MC variance of the 3M label, consistent with the learner's own error dominating. None of this shows that the two budgets are equivalent, nor that MC noise never matters beyond this learner and training scale.

## 6 Conclusion

A local balance can improve a completed Monte Carlo label, but its residual combines sampling noise with a persistent closure gap. For the classical linear correction, weighted by the MC covariance and a design model of the gap, a scene improves on average exactly when the variance removed exceeds the squared closure bias introduced. More samples shrink the noise but not the gap, so the same balance can help noisy labels and harm cleaner ones. When treated as exact, coarsening its equations cannot increase the variance removed. A one-label screen estimates a lower bound on expected gain on the scored support, without guaranteeing individual improvement.

Our experiments support these limits. All evaluated flat-terrain runs improved, and an oracle projection exposed nearly all the reference difference with cellwise equations but none with a domain-wide equation. Around buildings at 3M, class-median outcomes were negative in the three most stable classes. Above 9 m, most scenes worsened and screening poorly predicted stored-layer outcomes. On the stored layer, where corrections both helped and hurt, the run-averaged screen predicted $33/34$ signs, against $20/34$ for always predicting improvement. At 300k, correction improved $31/34$ labels on the scored stored-layer support. The screen also predicted $31/34$ signs correctly, matching always predicting improvement and accepting two harmful corrections. The auxiliary full-budget run increases simulation cost, while estimated covariances and parameter reuse limit the theoretical guarantees. Separately, reducing uncorrected-label variance tenfold did not meet our prespecified heuristic criterion for a learning benefit at the tested scale. This shows neither equivalence between budgets nor irrelevance of MC noise elsewhere.

The next step is to train on corrected labels, comparing against uncorrected-label training at the same original sample count and at the same total simulation cost, including auxiliary runs. Repeating these comparisons across training-set sizes would test when label noise becomes limiting. Applying only screen-accepted corrections would test whether screening improves the trained surrogate.

### AI USE STATEMENT

Generative-AI tools were used to edit and polish the manuscript for clarity and readability, to identify potentially relevant literature, and to draft or revise portions of the text, including the proofs, from notes, results and instructions provided by the authors. AI coding assistants were also used, under the authors' direction, to implement and run parts of the simulation, correction and analysis code and to generate figures. The authors checked the cited sources, the proofs and the reported numbers against the experiment outputs, and take full responsibility for the final content.

### ETHICS STATEMENT

This methodological study tests a way to reduce Monte Carlo noise in labels from a physical simulator, using pollutant dispersion around synthetic airports as its case study. Its dataset, AirDispNet (Appendix D), contains only synthetic scenes, with procedural airport geometry, sampled meteorology and emission inventories that follow published aviation methods. It contains no observations or operational records from real airports. The study involves no human participants and uses no personal or identifying data.

The learned surrogate is a research model, not a validated regulatory tool. In the setting tested here, reducing label noise did not detectably improve its accuracy. Use for environmental, regulatory or public-health decisions would require independent, application-specific validation. The authors declare no competing interests.

## REPRODUCIBILITY STATEMENT

Section 3 states the theoretical assumptions, with proofs in Appendix B. Appendix A details the numerical construction of the correction. Appendix C documents the experimental protocols, reference construction, evaluation supports, and sensitivity analyses, and Appendix C.5 gives the architecture and training configuration of the learner. Appendix D describes the dataset and frozen splits. Each training run records its full configuration, its training history and the SHA-256 digests of the frozen split and configuration files.

## REFERENCES

Alexander C. Aitken. On least squares and linear combination of observations. *Proceedings of the Royal Society of Edinburgh*, 55:42–48, 1935. doi: 10.1017/S0370164600014346.

Steve Bako, Thijs Vogels, Brian McWilliams, Mark Meyer, Jan Novák, Alex Harvill, Pradeep Sen, Tony DeRose, and Fabrice Rousselle. Kernel-predicting convolutional networks for denoising Monte Carlo renderings. *ACM Transactions on Graphics*, 36(4):97:1–97:14, 2017. doi: 10.1145/3072959.3073708.

M. E. Berlyand. *Contemporary Problems of Atmospheric Diffusion and Pollution of the Atmosphere*. Gidrometeoizdat, Leningrad, 1975. English translation, U.S. Environmental Protection Agency, Raleigh, NC.

Alberto Carrassi, Marc Bocquet, Laurent Bertino, and Geir Evensen. Data assimilation in the geosciences: An overview of methods, issues, and perspectives. *WIREs Climate Change*, 9(5):e535, 2018. doi: 10.1002/wcc.535.

Ayce Celikel, Ian Fuller, Marcel Silue, Serge Peeters, and Nicolas Duchene. Airport local air quality studies (ALAQS): Concept document. SEE Note EEC/SEE/2005/003, EUROCONTROL Experimental Centre, Brétigny-sur-Orge, France, 2005. Issue 2.1, dated 01/2006.

Cameron M. Crowe. Data reconciliation—progress and challenges. *Journal of Process Control*, 6 (2–3):89–98, 1996. doi: 10.1016/0959-1524(96)00012-1.

Daniel Hodouin, Azar Mirabedini, Sami Makni, and Claude Bazin. Reconciliation of mineral processing data containing correlated measurement errors. *International Journal of Mineral Processing*, 54(3–4):201–215, 1998. doi: 10.1016/S0301-7516(98)00012-X.

M. F. Hutchinson. A stochastic estimator of the trace of the influence matrix for Laplacian smoothing splines. *Communications in Statistics – Simulation and Computation*, 19(2):433–450, 1990.

Ingenieurbüro Janicke. Ausbreitungsmodell nach TA Luft – AUSTAL: Programmbeschreibung zu version 3.3. Technical report, Umweltbundesamt, Dessau-Roßlau, Germany, 2024. Stand 2024-03-22.

International Civil Aviation Organization. Annex 16 to the convention on international civil aviation – environmental protection, volume ii: Aircraft engine emissions. Technical report, ICAO, Montréal, Canada, 2017.

International Civil Aviation Organization. Airport air quality manual. Technical Report Doc 9889, ICAO, Montréal, Canada, 2020.

International Civil Aviation Organization. ICAO aircraft engine emissions databank. Hosted by the European Union Aviation Safety Agency (EASA), 2024. https://www.easa.europa.eu/en/domains/environment/icao-aircraft-engine-emissions-databank.

U. Janicke, E. Fleuti, and I. Fuller. LASPORT – a model system for airport-related source systems based on a Lagrangian particle model. In *Proc. 11th Int. Conf. on Harmonisation within Atmospheric Dispersion Modelling for Regulatory Purposes (HARMO)*, pp. 352–356, Cambridge, UK, 2007.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. In *International Conference on Learning Representations (ICLR)*, 2021a.

Zongyi Li, Hongkai Zheng, Nikola Kovachki, David Jin, Haoxuan Chen, Burigede Liu, Kamyar Azizzadenesheli, and Anima Anandkumar. Physics-informed neural operator for learning partial differential equations. *arXiv preprint arXiv:2111.03794*, 2021b.

Ning Liu, Yiming Fan, Xianyi Zeng, Milan Klöwer, Lu Zhang, and Yue Yu. Harnessing the power of neural operators with automatically encoded conservation laws. In *International Conference on Machine Learning (ICML)*, volume 235 of *PMLR*, pp. 30965–30997, 2024.

Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. *Nature Machine Intelligence*, 3(3):218–229, 2021. doi: 10.1038/s42256-021-00302-5.

Didier Maquin, Olivier Adrot, and José Ragot. Data reconciliation with uncertain models. *ISA Transactions*, 39(1):35–45, 2000. doi: 10.1016/S0019-0578(99)00045-2.

Rafid Umayer Murshed, Saif Ur Rahman, Mingyue Tang, and Elahe Soltanaghai. Physics-unrolled neural operator for wireless field modeling. *arXiv preprint arXiv:2608.18495*, 2026. doi: 10.48550/arXiv.2608.18495. URL https://arxiv.org/abs/2608.18495.

Art B. Owen. *Monte Carlo Theory, Methods and Examples*. Published online by the author, 2013. https://artowen.su.domains/mc/.

Matthew J. Picklo, Qi Tang, Yanzeng Zhang, Jennifer K. Ryan, and Xian-Zhu Tang. Denoising particle-in-cell data via smoothness-increasing accuracy-conserving filters with application to Bohm speed computation. *Journal of Computational Physics*, 502:112790, 2024. doi: 10.1016/j.jcp.2024.112790.

Charles M. Stein. Estimation of the mean of a multivariate normal distribution. *The Annals of Statistics*, 9(6):1135–1151, 1981.

G. I. Taylor. Diffusion by continuous movements. *Proceedings of the London Mathematical Society*, s2-20(1):196–212, 1922.

D. J. Thomson. Criteria for the selection of stochastic models of particle trajectories in turbulent flows. *Journal of Fluid Mechanics*, 180:529–556, 1987.

Hrishikesh Viswanath, Hong Chul Nam, Xi Deng, Julius Berner, Anima Anandkumar, and Aniket Bera. Operator learning using weak supervision from walk-on-spheres. *arXiv preprint arXiv:2603.01193*, 2026. doi: 10.48550/arXiv.2603.01193. URL https://arxiv.org/abs/2603.01193.

Yuqi Wei, Fang Chen, Yingxuan Qin, Liaolei He, and Yang Wang. Physics-informed and noise-assisted self-supervised learning with dual samples input for hypersonic DSMC denoising. *Acta Astronautica*, 241:153–169, 2026. doi: 10.1016/j.actaastro.2025.12.056.

Matthias Zwicker, Wojciech Jarosz, Jaakko Lehtinen, Bochang Moon, Ravi Ramamoorthi, Fabrice Rousselle, Pradeep Sen, Cyril Soler, and Sung-Eui Yoon. Recent advances in adaptive sampling and reconstruction for Monte Carlo rendering. *Computer Graphics Forum*, 34(2):667–681, 2015. doi: 10.1111/cgf.12592.

## A Estimator details and implementation

Section 4 builds the balance $L$, the MC covariance $\Sigma_N$ and the trust weights $\Sigma_\delta$, then corrects and screens a label. This appendix gives the numerical choices needed to reproduce these steps.

**The balance $L$ and the source $q$.** The simulator's grid has square horizontal cells 5 m wide and 42 layers, 3 m thick up to 87 m and thicker above, up to 1,500 m. $L$ discretizes the flux divergence $\nabla \cdot (u\,c - \kappa \nabla c)$ on this grid. Each first derivative is a central difference, the difference between the two neighbouring cells divided by their distance. On the vertical axis, a three-point formula accounts for the unequal thickness of the layers. One difference gives the gradient $\nabla c$ in the flux, and another the divergence of the flux, so each equation involves a cell and nearby cells up to two cells away. The wind $u$ is the simulator's own three-dimensional wind field on the grid, which flows around the buildings when there are any. The diffusivity $\kappa$ starts from the along-wind, cross-wind and vertical diffusivities $K_u$, $K_v$ and $K_w$ of the scene's turbulence profile, which vary only with height. In each cell, with $\theta$ the angle of the local horizontal wind measured from the grid's $x$ axis, $\kappa_{xx} = K_u \cos^2\theta + K_v \sin^2\theta$, $\kappa_{yy} = K_u \sin^2\theta + K_v \cos^2\theta$, $\kappa_{xy} = \kappa_{yx} = (K_u - K_v)\sin\theta\cos\theta$ and $\kappa_{zz} = K_w$, the other entries being zero. The building-wake diffusivity, the extra mixing behind buildings, is then added to $\kappa_{xx}$, $\kappa_{yy}$ and $\kappa_{zz}$. The simulator adds it in the same way, but only where it exceeds 5% of $K_w$ or where the standard deviation of the wake's velocity exceeds 5% of that of the background vertical velocity. Building cells carry no equation and enter none. Next to a building or a face of the domain, a derivative uses a one-sided difference toward the inside of the flow, so no equation draws on a building cell. The source vector $q$ holds, in the cell containing each source point, that source's emission rate divided by the cell volume, and zero elsewhere, so that $Lc$ and $q$ are both in g m$^{-3}$ s$^{-1}$. For PM-1 the rate is the PM-1 share of the source's particulate emission. The label run reads the wind stored with the full-budget run instead of solving for it again, so both runs share the same $L$.

**The correlations of $\Sigma_N$.** Section 4.1 writes $\Sigma_N = \mathrm{diag}(\sigma)\, R\, \mathrm{diag}(\sigma)$ and takes the standard deviations $\sigma_i$ of the label from the full-budget run's reported relative errors (Equation (17)). The correlation matrix $R$ has one block per horizontal layer and treats cells of different layers as uncorrelated. On layer $z$,

$$R_z = (1-\beta_z)\, I + \beta_z\, A_z, \qquad A_z = \frac{k_z \star k_z}{(k_z \star k_z)(0)}, \qquad k_z(h) = \exp\Big( - \tfrac{|h\cdot e_\parallel|}{\lambda_{\parallel,z}} - \tfrac{|h\cdot e_\perp|}{\lambda_{\perp,z}} \Big), \quad (9)$$

with $\beta_z \in [0,1]$. At $\beta_z = 0$ the cell errors are uncorrelated, and a larger $\beta_z$ gives more weight to the correlated part $A_z$. The kernel $k_z$ weights each lag $h$, the displacement between two cells of the layer counted in cells. The unit vectors $e_\parallel$ and $e_\perp$ point along and across the scene's prescribed wind direction, the same for all layers, unlike the local wind used in $L$. The kernel decays exponentially in both directions, over its along-wind and cross-wind decay lengths $\lambda_{\parallel,z}$ and $\lambda_{\perp,z}$. In practice $k_z$ is cut to a square of half-width $\max(2, \lceil 4\lambda_{\max} \rceil)$ cells, with $\lambda_{\max}$ the largest decay length over the scene's layers. The star $\star$ is the discrete convolution. Since $k_z$ is symmetric, $(k_z \star k_z)(h) = \sum_{h'} k_z(h')\, k_z(h - h')$ is its autocorrelation at lag $h$, and dividing by its value at $h = 0$ makes it equal to 1 there. Entry $(A_z)_{ij}$ is this normalized autocorrelation at the lag between cells $i$ and $j$. The correlation between two cells thus follows the autocorrelation of $k_z$, not $k_z$ itself, so the decay

lengths are kernel parameters rather than correlation lengths. $(A_z)_{ij}$ is also the inner product of $g_i$ and $g_j$, where $g_i$ is the cut kernel centred on cell $i$, divided by its norm and extended by zeros, even beyond the grid. Hence $v^\top A_z v = \| \sum_i v_i g_i \|^2 \geq 0$ for every vector $v$, so $A_z$ is positive semidefinite with unit diagonal. $R_z$, a weighted average of $I$ and $A_z$, is thus a valid correlation matrix without further adjustment. The product of $R_z$ with a vector is computed by a fast Fourier transform (FFT) convolution with zero padding, so opposite edges of the domain are never joined.

The fit targets correlations between cells, estimated across the ten particle groups of the label run. Multiplying the standard error that the run reports for a cell by $\sqrt{10}$ estimates the standard deviation of one group result there. Dividing group $g$'s deviation from the run's mean in cell $i$ by this quantity gives the normalized deviation $\eta_{g,i}$. Only cells with a positive reported error, a concentration of at least $10^{-6}\,\mathrm{g\,m^{-3}}$ and at least $10^{-3}$ of the layer's maximum in the label run enter a layer's estimate. At lag $h$, the estimated correlation is

$$\widehat{\rho}(h) = \frac{\langle \eta_{g,i}\, \eta_{g,j} \rangle}{\frac{1}{2}\langle \eta_{g,i}^2 + \eta_{g,j}^2 \rangle},$$

where $\langle \cdot \rangle$ averages over the ten groups and over all pairs $(i, j)$ of selected cells separated by $h$. We use lags of one to six cells along the grid's $x$ axis and one to three along $y$, each only if at least 100 pairs contribute. At a lag $h \neq 0$ the model's correlation is $\beta_z$ times the normalized autocorrelation $A_z$ at $h$, with the kernel oriented along the wind. The fit tries 16 values of $\lambda_{\parallel,z}$ between 0.5 and 12 cells and 12 values of $\lambda_{\perp,z}$ between 0.1 and 4 cells, spaced geometrically, with $\lambda_{\parallel,z} \geq \lambda_{\perp,z}$. For each pair, $\beta_z$ takes its least-squares value, kept in $(0, 1]$. The pair with the smallest root-mean-square misfit over the usable lags is refined twice on finer grids. A layer with fewer than 200 selected cells, or with fewer than three usable lags along either axis, takes the parameters of the nearest fitted layer. Under the scaling $\Sigma_N = \Sigma_1 / N$ of Section 3.1, the correlations do not depend on $N$, so we reuse a fitted model across budgets.

For $\mathrm{NO}_x$ this fit reads the label's own run, for the correlations and for the choice of cells, so $R$ depends on the label, whereas Section 3 treats $\Sigma_N$ as fixed in advance. The experiments of Section 5 build $\Sigma_N$ differently. For the flat scenes at 250k particles, $\sigma_i^2$ is the sample variance of cell $i$ over the eight 250k runs, which include the scored run. $R$ is fitted to the correlations across the eight 2M-particle runs of the scene. There, each run's deviation from the mean of the eight is normalized by their sample standard deviation, cells are selected where this sample standard deviation is positive and the 12M reference exceeds $10^{-3}$ of its three-dimensional maximum, and a lag needs 60 pairs. For the airport scenes at 3M particles, $\sigma_i$ is the eight runs' mean reported relative error times their mean field, and $R$ is the fit of the scene's separate 300k run. On airport scenes, PM-1 always uses the $\mathrm{NO}_x$ fit of the same scene, from a separate transport run, which assumes that deposition leaves the correlations unchanged.

**The trust weights $\Sigma_\delta$.** In Equation (7), the maximum with zero is taken once, after the noise term is subtracted. The noise term is $(N/N_{\mathrm{full}})\, d_i$, with $d_i = [L\,\mathrm{diag}(\sigma^2) L^\top]_{ii} = \sum_j L_{ij}^2 \sigma_j^2$. Here $d_i$ is the variance that the label's MC noise gives to equation $i$'s residual, predicted as if cells erred independently, and the factor $N/N_{\mathrm{full}}$ scales it to the full-budget run. All $d_i$ together cost one multiplication by the matrix of squared entries of $L$. The floor $\gamma$ is $10^{-3}$ times the median of $d_i$ over the field above 9 m. Now $d_i > 0$ as soon as one cell of equation $i$ has $\sigma_j > 0$. A scored cell has $c_i^{\mathrm{full}} > 0$, so its $\sigma_i$ vanishes only if its ten group doses are exactly equal (Equation (17)). Hence $\gamma = 0$ would require this in every cell of at least half of the scored equations. Otherwise $\gamma > 0$, and $\Sigma_\delta$ and $\Sigma_r$ are positive definite. The eight-run airport experiments replace the full-budget run by their eight runs. An entry is then the square of the mean of $(Lc)_i$ over the runs, minus the variance of that mean, the sample variance of $(Lc)_i$ across runs divided by eight. Their $\gamma$ is $10^{-3}$ times the median of that sample variance over the field above 9 m, which is zero only if the eight runs give exactly equal $(Lc)_i$ in at least half of the scored equations. Flat scenes form the same difference for each of their four eight-run sets, at 250k, 500k, 1M and 2M particles. Each difference is an unbiased estimate of the same squared mean, since $c^\star$ does not depend on $N$, and we average the four before taking the maximum with zero. Their $\gamma$ uses the sample variance of the 2M runs. Every setting squares the net outflow rather than the residual. This lowers the trust in a source cell's equation when the emission dominates its balance.

Two of these choices push $\Sigma_\delta$ in opposite directions. The maximum with zero can only raise entries, and a larger $\Sigma_\delta$ can only lower $\phi_{\mathrm{design}}$ (Remark 2). The independent-cell noise term ignores the

correlations of $\Sigma_N$. Summed over the field above 9 m and measured with random probes on six deployment scenes, it overstates the subtracted variance by a factor of $1.05$ to $1.64$, the largest in a very stable scene. Since the full-budget run carries a tenth of the label's noise, the excess is only $0.5$ to $6.4\%$ of the label's noise variance in the residual, summed over the same cells.

**Supports, solve and traces.** The *field above* $9\,m$ keeps the in-air cells where the full-budget field exceeds $10^{-3}$ of its three-dimensional maximum, that share no face with a building, and that lie at least three cells from every face of the domain. This margin keeps the one-sided differences of the domain faces out of the scored cells' equations. It removes three layers at each end, so the field runs from 9 to 800 m. The *stored-layer support* applies the same criteria to the lowest layer, with the margin on the lateral faces only. On flat scenes, both supports also leave out a box of $7 \times 7 \times 7$ cells centred on the source. The field that sets the threshold is the full-budget run in deployment, the mean of the eight runs on airport scenes and the 12M reference on flat scenes.

The correction solves $\Sigma_r\, y = r$, with $\Sigma_r = L\Sigma_N L^\top + \Sigma_\delta$ and $r = L\tilde{c} - q$, and sets $\hat{c} = \tilde{c} - \Sigma_N L^\top y$. Conjugate gradients never form $\Sigma_r$. Each product $\Sigma_r v$ applies $L^\top$, then $\mathrm{diag}(\sigma)$, $R$ by FFT and $\mathrm{diag}(\sigma)$ again, then $L$, and adds $\Sigma_\delta v$. The iterations stop once $\|\Sigma_r y - r\| \le 10^{-6}\|r\|$. The preconditioner, a rescaling that speeds convergence, divides equation $i$ by $d_i + (\Sigma_\delta)_{ii}$, the diagonal of $L\,\mathrm{diag}(\sigma^2)L^\top + \Sigma_\delta$. Solves take a few hundred iterations, at most 1,376 on replicate scenes.

Write $\mathrm{tr}_M$ for the sum of the diagonal entries of the scored cells. The design-predicted gain $\phi_{\mathrm{design}} = \mathrm{tr}_M(K^\star L\Sigma_N)/D_M$ and the noise term $\tau = \mathrm{tr}_M(K^\star L\Sigma_N L^\top K^{\star\top})$ involve traces of matrices that are never formed. The Hutchinson estimator (Hutchinson, 1990) draws probe vectors $z$ whose entries are independent, $+1$ or $-1$ with probability $1/2$ on the scored cells, and 0 elsewhere. Then $\mathbb{E}[z_i z_j]$ is 1 for $i = j$ scored and 0 otherwise, so $\mathbb{E}[z^\top A z] = \mathrm{tr}_M A$, and averaging $z^\top A z$ over probes estimates the trace without bias. For each probe, one solve with $\Sigma_r$ gives both quadratic forms. The deployment and the replicate scenes use 24 probes, the flat scenes 12. On three deployment scenes, in classes I, III/1 and V, the relative standard error of the probe average is $0.4$ to $1.7\%$ on $\phi_{\mathrm{design}}$ and $0.4$ to $1.9\%$ on $\tau$, or $0.003$ to $0.010$ on $\phi_{\mathrm{design}}$ itself. On the six flat scenes it is $0.2$ to $1.3\%$ on $\phi_{\mathrm{design}}$. These errors come from the random probes only, not from the estimated covariance or the realized gain.

## B Proofs of the theoretical results

This appendix proves the results of Section 3 in their order, with the notation of that section. Each result is restated before its proof.

### B.1 Notation and facts used

**Traces and squared errors.** For a matrix $A$, $A^\top$ is its transpose, $(A^\top)_{ij} = A_{ji}$, and $A$ is symmetric when $A = A^\top$. Covariances and second moments are symmetric, since their $(i, j)$ and $(j, i)$ entries are the same expectation. So is $B\Sigma B^\top$ for a symmetric $\Sigma$, since $(B\Sigma B^\top)^\top = B\Sigma^\top B^\top$. The trace $\mathrm{tr}\, A = \sum_i A_{ii}$ of a square matrix is linear, and $\mathrm{tr}(AB) = \mathrm{tr}(BA)$ whenever both products are square. In particular, $\|x\|^2 = x^\top x = \mathrm{tr}(xx^\top)$ for a vector $x$. For a random vector $x$ and a fixed matrix $A$, the expectation passes inside the trace, since both are linear,

$$\mathbb{E}\|Ax\|^2 = \mathbb{E}\,\mathrm{tr}\left(Axx^\top A^\top\right) = \mathrm{tr}\left(A\,\mathbb{E}[xx^\top]\,A^\top\right). \tag{10}$$

This turns every expected squared error below into a trace.

**Positive semidefinite matrices.** A symmetric matrix $A$ is positive semidefinite, $A \succeq 0$, when $x^\top A x \ge 0$ for every vector $x$, and positive definite, $A \succ 0$, when $x^\top A x > 0$ for every $x \ne 0$. A positive definite $A$ is invertible, since $Ax = 0$ gives $x^\top A x = 0$ and hence $x = 0$. Its inverse is symmetric, since $(A^{-1})^\top = (A^\top)^{-1}$, and positive definite, since $x^\top A^{-1} x = y^\top A y$ with $y = A^{-1}x$. We write $A \succeq B$ when $A - B \succeq 0$, an order between symmetric matrices called the Loewner order. It is partial, since two symmetric matrices need not be comparable. The proofs use four facts.

(a) If $S \succeq 0$, then $VSV^\top \succeq 0$ for every matrix $V$ of matching size, since $x^\top VSV^\top x = (V^\top x)^\top S\,(V^\top x) \ge 0$. If $S \succ 0$ and the rows of $V$ are linearly independent, then $V^\top x \ne 0$ for $x \ne 0$, so $VSV^\top \succ 0$.

(b) If $A \succeq 0$, its diagonal entries $A_{ii} = e_i^\top A e_i$, with $e_i$ the $i$-th unit vector, are nonnegative, so $\operatorname{tr} A \geq 0$.

(c) If $A \succ 0$ and $y_k^\top$ is row $k$ of a matrix $Y$, then $\operatorname{tr}(YAY^\top) = \sum_k y_k^\top A y_k$. Each term is positive unless $y_k = 0$, so this trace is positive unless $Y = 0$.

(d) A symmetric $A$ has an orthonormal basis of eigenvectors $u_k$, with real eigenvalues $\lambda_k$, and $A = \sum_k \lambda_k u_k u_k^\top$ (the spectral theorem). If $A \succeq 0$, then $\lambda_k = u_k^\top A u_k \geq 0$. For $M \succeq 0$ with eigenvalues $\mu_k$ and eigenvectors $v_k$, and for $A \succeq 0$, it follows that $\operatorname{tr}(MA) = \sum_k \mu_k v_k^\top A v_k \geq 0$.

**The model.** As in Section 3.1, a run returns $\tilde{c} = c^\star + \varepsilon$, where $c^\star = \mathbb{E}_\varepsilon \tilde{c}$, $\mathbb{E}_\varepsilon \varepsilon = 0$ and $\Sigma_N = \mathbb{E}_\varepsilon[\varepsilon\varepsilon^\top]$. Whether $c^\star$ is physically true plays no role. The balance gives $Lc^\star = q + \bar{\delta}$ and the residual $r = L\tilde{c} - q = \bar{\delta} + L\varepsilon$. Two averages occur, and each result states which one it uses. On a *fixed scene*, the gap is the fixed vector $\bar{\delta}$, and $\mathbb{E}_\varepsilon$ averages over runs. This is the setting of Proposition 2 and Corollary 1. Under the *design model* of Assumption 2, the gap is a random vector $\delta$ with $\mathbb{E}_{\varepsilon,\delta}[\varepsilon\delta^\top] = 0$, and $r = \delta + L\varepsilon$. This is the setting of Theorem 1, and it serves only to choose $K$. Only this absence of correlation is used, not independence. $\Sigma_\delta = \mathbb{E}_\delta[\delta\delta^\top]$ is a second moment, not a covariance, so $\delta$ may have a nonzero mean. The matrices $L$, $K$, $\Sigma_N$ and $\Sigma_\delta$, the vector $q$ and the supports below are chosen without using the label. Hence $K^\star$, $D$, $\phi_{\text{design}}$, $\tau$ and $\tau_\delta$, built from them, are fixed when we average over runs. This is how the proofs use that condition.

**Positive definiteness of $\Sigma_N$.** Assumption 1 asks $\Sigma_N \succ 0$. The proofs of Theorem 1, Proposition 1(i), Proposition 2 and Corollary 1 use only $\Sigma_N \succeq 0$, $\Sigma_r \succ 0$ and $D > 0$. Proposition 1(ii)–(iv) use $\Sigma_N \succ 0$ to obtain $\Sigma_r = L\Sigma_N L^\top \succ 0$ from linearly independent equations, by fact (a). This is the extension stated in Section 3.1.

**Derived quantities.** With $\Sigma_r = L\Sigma_N L^\top + \Sigma_\delta \succ 0$, Section 3 defines

$$K^\star = \Sigma_N L^\top \Sigma_r^{-1}, \qquad \Sigma_{\text{post}} = \Sigma_N - K^\star L \Sigma_N, \qquad D = \operatorname{tr} \Sigma_N, \qquad \phi_{\text{design}} = \frac{D - \operatorname{tr} \Sigma_{\text{post}}}{D},$$

$$b = \|K^\star \bar{\delta}\|^2, \qquad \tau = \operatorname{tr}\left(K^\star L \Sigma_N L^\top K^{\star\top}\right), \qquad \tau_\delta = \operatorname{tr}\left(K^\star \Sigma_\delta K^{\star\top}\right), \qquad \phi_{\text{net}} = \phi_{\text{design}} - \frac{b}{D}.$$

By facts (a) and (b), $\tau \geq 0$ and $\tau_\delta \geq 0$. Multiplying the definition of $K^\star$ on the right by $\Sigma_r$ gives $K^\star \Sigma_r = \Sigma_N L^\top$. Transposing, with $\Sigma_N$ and $\Sigma_r$ symmetric, gives

$$\Sigma_r K^{\star\top} = L \Sigma_N. \tag{11}$$

Since $\Sigma_r = L\Sigma_N L^\top + \Sigma_\delta$, linearity of the trace and (11) give the identity used in Section 3.4,

$$\tau + \tau_\delta = \operatorname{tr}\left(K^\star \Sigma_r K^{\star\top}\right) = \operatorname{tr}\left(K^\star L \Sigma_N\right) = \operatorname{tr}\left(\Sigma_N - \Sigma_{\text{post}}\right) = D\,\phi_{\text{design}}. \tag{12}$$

**Supports.** A support is a set of cells chosen without using the label. Its matrix $M$ is diagonal, with $M_{ii} = 1$ on the support and 0 elsewhere. We write $\|x\|_M^2 = x^\top M x$ and $\operatorname{tr}_M A = \operatorname{tr}(MA)$, the sum of the diagonal entries of $A$ on the support, and assume $D_M = \operatorname{tr}_M \Sigma_N > 0$. Replacing $\|\cdot\|^2$, tr and $D$ by $\|\cdot\|_M^2$, $\operatorname{tr}_M$ and $D_M$ in the definitions above gives $b$, $\tau$, $\tau_\delta$, $\phi_{\text{design}}$ and $\phi_{\text{net}}$ on the support, for which we keep the same symbols. Only the sums are restricted, not $K^\star$ or $\Sigma_{\text{post}}$. The proofs only take traces of fixed matrices and expectations of squared norms. They stay valid on a support, since (10) becomes $\mathbb{E}\|Ax\|_M^2 = \operatorname{tr}_M(A\,\mathbb{E}[xx^\top]A^\top)$, fact (b) gives $\operatorname{tr}_M A \geq 0$ for $A \succeq 0$, and (12) holds with $\operatorname{tr}_M$ and $D_M$ by the same steps. Each proof says where it needs more.

### B.2 Proof of Theorem 1

**Theorem 1 (restated).** *Under Assumptions 1–2, suppose $\Sigma_r \succ 0$, as holds when $\Sigma_\delta \succ 0$, and let $K^\star := \Sigma_N L^\top \Sigma_r^{-1}$ and $\Sigma_{\text{post}} := \Sigma_N - K^\star L \Sigma_N$. For every fixed $K$, $\Gamma(K) = \Sigma_{\text{post}} + (K - K^\star)\Sigma_r(K - K^\star)^\top$. The trace of the last term is positive unless $K = K^\star$. Hence $K^\star$ is the unique minimizer of the expected squared error among corrections $\tilde{c} - Kr$, $\Gamma(K^\star) = \Sigma_{\text{post}}$, and the corrected label is $\hat{c} := \tilde{c} - \Sigma_N L^\top (L\Sigma_N L^\top + \Sigma_\delta)^{-1}(L\tilde{c} - q)$.*

**Lemma 1** (Completion of squares). *Suppose* $\Sigma_N \succeq 0$, $\Sigma_\delta \succeq 0$ *and* $\Sigma_r \succ 0$, *and let* $\Gamma(K) := (I - KL)\Sigma_N(I - KL)^\top + K\Sigma_\delta K^\top$ *for* $K \in \mathbb{R}^{n_c \times n_r}$. *Then*

$$\Gamma(K) = \Sigma_{\text{post}} + (K - K^\star)\,\Sigma_r\,(K - K^\star)^\top. \tag{13}$$

*Proof.* Since $(I - KL)^\top = I - L^\top K^\top$, expanding the first product of $\Gamma(K)$ and grouping the two terms quadratic in $K$ with $L\Sigma_N L^\top + \Sigma_\delta = \Sigma_r$ gives

$$\Gamma(K) = \Sigma_N - KL\Sigma_N - \Sigma_N L^\top K^\top + K\Sigma_r K^\top. \tag{14}$$

The last term of (13) expands into four terms. By (11), $K\Sigma_r K^{\star\top} = KL\Sigma_N$, its transpose is $K^\star\Sigma_r K^\top = \Sigma_N L^\top K^\top$, and $K^\star\Sigma_r K^{\star\top} = K^\star L\Sigma_N$. Hence

$$(K - K^\star)\Sigma_r(K - K^\star)^\top = K\Sigma_r K^\top - KL\Sigma_N - \Sigma_N L^\top K^\top + K^\star L\Sigma_N.$$

Adding $\Sigma_{\text{post}} = \Sigma_N - K^\star L\Sigma_N$ cancels the last term and gives the right side of (14). □

*Proof of Theorem 1.* Fix $K$. Under the design model, $r = \delta + L\varepsilon$, so the corrected label has error

$$e := \hat{c}_K - c^\star = \varepsilon - K(\delta + L\varepsilon) = (I - KL)\varepsilon - K\delta. \tag{15}$$

Expanding $ee^\top$ gives $(I - KL)\varepsilon\varepsilon^\top(I - KL)^\top + K\delta\delta^\top K^\top$ and two cross terms. Since $K$ is fixed, the first two terms have means $(I - KL)\Sigma_N(I - KL)^\top$ and $K\Sigma_\delta K^\top$, and the cross terms have zero mean, since $\mathbb{E}_{\varepsilon,\delta}[\varepsilon\delta^\top] = 0$ by Assumption 2. Hence $\mathbb{E}_{\varepsilon,\delta}[ee^\top] = \Gamma(K)$, which is (3), and $\mathbb{E}_{\varepsilon,\delta}\|e\|^2 = \operatorname{tr}\Gamma(K)$. By Lemma 1, $\operatorname{tr}\Gamma(K) = \operatorname{tr}\Sigma_{\text{post}} + \operatorname{tr}\left((K - K^\star)\Sigma_r(K - K^\star)^\top\right)$, and by fact (c) the last trace is positive unless $K = K^\star$. So $K^\star$ is the unique minimizer of the expected squared error, and $\Gamma(K^\star) = \Sigma_{\text{post}}$. Substituting $K^\star$ into $\hat{c} = \tilde{c} - K^\star r$ gives (5). When $\Sigma_\delta \succ 0$, fact (a) gives $x^\top\Sigma_r x \geq x^\top\Sigma_\delta x > 0$ for $x \neq 0$, so $\Sigma_r \succ 0$. □

The proof shows that $\operatorname{tr}\Sigma_{\text{post}}$ is the expected squared error after correction, as Definition 1 states. Since $r = \delta + L\varepsilon$, Assumption 2 also gives $\mathbb{E}_{\varepsilon,\delta}[\varepsilon r^\top] = \Sigma_N L^\top$ and $\mathbb{E}_{\varepsilon,\delta}[rr^\top] = L\Sigma_N L^\top + \Sigma_\delta = \Sigma_r$, as used after the theorem. By fact (a), the last term of (13) is positive semidefinite, so $\Gamma(K) \succeq \Gamma(K^\star)$ for every $K$. $K^\star$ thus minimizes the whole second moment in the Loewner order, hence it is a minimizer of every weighted risk $\operatorname{tr}(M\,\Gamma(K))$ with $M \succeq 0$, by fact (d). Uniqueness can fail for a singular $M$. On a support, for instance, $\operatorname{tr}_M\left((K - K^\star)\Sigma_r(K - K^\star)^\top\right)$ vanishes whenever the rows of $K$ for the scored cells equal those of $K^\star$, and the other rows do not affect the score. If moreover $\mathbb{E}_\delta\delta = 0$, (15) gives $\mathbb{E}_{\varepsilon,\delta}e = 0$ for every $K$. $\Gamma(K)$ is then the covariance of $e$, and $K^\star$ has the smallest total variance, as in generalized least squares (Aitken, 1935).

### B.3 Proof of Proposition 1

**Proposition 1 (restated).** *In the setting of Theorem 1, (i)* $\phi_{\text{design}} \in [0, 1]$. *Suppose now that the design treats the balance as exact,* $\Sigma_\delta = 0$, *and that the equations of each balance are linearly independent, so that* $\Sigma_r \succ 0$. *Then (ii) two balances with the same row space give the same* $\phi_{\text{design}}$; *(iii) if* $L_{\text{coarse}} = AL_{\text{fine}}$ *for a matrix A, so that each coarse equation sums multiples of fine ones, then* $\phi_{\text{design}}(L_{\text{coarse}}) \leq \phi_{\text{design}}(L_{\text{fine}})$; *and (iv) if all entries have uncorrelated errors of equal variance, a single equation gives* $\phi_{\text{design}} = 1/n_c$.

*Proof. (i)* $\Sigma_{\text{post}} = \Gamma(K^\star)$ is a sum of two matrices of the form $VSV^\top$ with $S \succeq 0$. It is thus positive semidefinite by fact (a), and $\operatorname{tr}\Sigma_{\text{post}} \geq 0$ by fact (b). Hence $\phi_{\text{design}} = 1 - \operatorname{tr}\Sigma_{\text{post}}/D \leq 1$. Next, $\Sigma_N - \Sigma_{\text{post}} = K^\star L\Sigma_N = \Sigma_N L^\top\Sigma_r^{-1}L\Sigma_N$, with $\Sigma_r^{-1} \succ 0$. Fact (a), with $V = \Sigma_N L^\top$, makes it positive semidefinite, and fact (b) gives $\operatorname{tr}(\Sigma_N - \Sigma_{\text{post}}) \geq 0$, that is $\phi_{\text{design}} \geq 0$. Both steps hold on a support.

We prove (iii) before (ii), since (ii) follows by applying (iii) in both directions.

*(iii)* Let $L_{\text{coarse}} = AL_{\text{fine}}$, and mark with the same subscripts the quantities of each balance, both with $\Sigma_\delta = 0$ and $\Sigma_r \succ 0$. With $\Sigma_\delta = 0$, $\Gamma(K) = (I - KL)\Sigma_N(I - KL)^\top$ depends on $K$ only through $KL$. For every $K$, $KL_{\text{coarse}} = (KA)L_{\text{fine}}$, so

$$\Gamma_{\text{coarse}}(K) = \Gamma_{\text{fine}}(KA).$$

Every coarse correction thus has the error second moment of a fine one. Lemma 1 and fact (a) give $\Gamma_{\text{fine}}(K') \succeq \Sigma_{\text{post,fine}}$ for every $K'$. Hence $\Sigma_{\text{post,coarse}} = \Gamma_{\text{coarse}}(K^\star_{\text{coarse}}) = \Gamma_{\text{fine}}(K^\star_{\text{coarse}}A) \succeq \Sigma_{\text{post,fine}}$. By fact (b), $\operatorname{tr}\Sigma_{\text{post,coarse}} \geq \operatorname{tr}\Sigma_{\text{post,fine}}$, and the two balances share $D$, so $\phi_{\text{design}}(L_{\text{coarse}}) \leq \phi_{\text{design}}(L_{\text{fine}})$. The same holds with $\operatorname{tr}_M$ and $D_M$.

*(ii)* If $L_1$ and $L_2$ have the same row space, each row of $L_1$ is a combination of the rows of $L_2$. Stacking the coefficients as the rows of a matrix $S$ gives $L_1 = SL_2$, and likewise $L_2 = TL_1$. Applying (iii) in both directions, also on a support, gives $\phi_{\text{design}}(L_1) \leq \phi_{\text{design}}(L_2) \leq \phi_{\text{design}}(L_1)$.

*(iv)* Uncorrelated errors of equal variance give $\Sigma_N = \sigma^2 I$, with $\sigma > 0$ since $D > 0$, and $D = n_c\sigma^2$. A single equation is a row $w^\top$, and its linear independence means $w \neq 0$. Then $\Sigma_r = \sigma^2\|w\|^2$ is a positive number, $K^\star = w/\|w\|^2$ and $K^\star L\Sigma_N = \sigma^2 ww^\top/\|w\|^2$, whose trace is $\sigma^2$. Hence $\phi_{\text{design}} = \sigma^2/(n_c\sigma^2) = 1/n_c$. □

Can one equation remove more than $1/n_c$ when errors are correlated? It can, up to a limit set by the largest eigenvalue of $\Sigma_N$. For $\Sigma_N \succeq 0$ and $w^\top\Sigma_N w > 0$, the steps of (iv) give $K^\star = \Sigma_N w/(w^\top\Sigma_N w)$ and, by $\operatorname{tr}(AB) = \operatorname{tr}(BA)$,

$$\phi_{\text{design}}(w^\top) = \frac{w^\top\Sigma_N^2 w}{(w^\top\Sigma_N w)\,D} \leq \frac{\lambda_{\max}(\Sigma_N)}{D} = \frac{1}{r_{\text{eff}}(\Sigma_N)}.$$

Here $\lambda_{\max}(\Sigma_N)$ is the largest eigenvalue of $\Sigma_N$, and $r_{\text{eff}}(\Sigma_N) := \operatorname{tr}\Sigma_N/\lambda_{\max}(\Sigma_N)$ is its *effective rank*, equal to $n_c$ for $\sigma^2 I$. The inequality follows from fact (d). With $\Sigma_N = \sum_k \lambda_k u_k u_k^\top$ and $\lambda_k \geq 0$, $w^\top\Sigma_N^2 w = \sum_k \lambda_k^2 (u_k^\top w)^2 \leq \lambda_{\max}\sum_k \lambda_k (u_k^\top w)^2 = \lambda_{\max}\, w^\top\Sigma_N w$, with equality when $w$ is an eigenvector for $\lambda_{\max}$. The largest share that one equation can remove is thus $1/r_{\text{eff}}(\Sigma_N)$, which is at least $1/n_c$ since $\lambda_{\max}$ is at least the mean eigenvalue $D/n_c$. A given equation can remove less than $1/n_c$, since its gain depends on how its row lies relative to the eigenvectors of $\Sigma_N$. For two entries of unit variance with correlation $0.8$, their sum removes $0.9$ and their difference $0.1$, against $1/2$ for uncorrelated entries. The row $w = \mathbf{1}$ sums all entries of the label. Total mass would weight each cell by its volume. The sum of all cell equations, $w = L_{\text{cell}}^\top\mathbf{1}$, gives the one equation for the whole domain of Section 5.1.

*Remark* 1 (What an exact balance removes). Take $\Sigma_\delta = 0$, $\Sigma_N \succ 0$ and linearly independent equations. Under the design model, $\delta$ then vanishes with probability one, since $\|\delta\|^2 \geq 0$ has mean $\operatorname{tr}\Sigma_\delta = 0$. The error of the corrected label is thus $(I - K^\star L)\varepsilon$, of mean zero, so $\Gamma(K^\star)$ is its covariance and $\phi_{\text{design}}$ the fraction of MC variance removed, as stated after Proposition 1. Moreover $LK^\star = L\Sigma_N L^\top(L\Sigma_N L^\top)^{-1} = I$. The corrected label satisfies the balance exactly, $L\hat{c} = L\tilde{c} - LK^\star r = q$, and the error $(I - K^\star L)\varepsilon$ lies in the null space of $L$, since $L(I - K^\star L) = 0$. The correction removes the noise that the balance sees and keeps the part it cannot see. With $n_c$ independent equations, $L$ is square and invertible, so $\hat{c} = L^{-1}q$, as stated in Section 3.2.

The removed part is an orthogonal projection in suitable coordinates. Let $W := \sum_k \sqrt{\lambda_k}\, u_k u_k^\top$, built from the eigenvalues and eigenvectors of $\Sigma_N$ as in fact (d). $W$ is symmetric and invertible, with $W^2 = \Sigma_N$. The whitened noise $z := W^{-1}\varepsilon$ has $\mathbb{E}_\varepsilon[zz^\top] = W^{-1}\Sigma_N W^{-1} = I$, that is, uncorrelated entries of unit variance. With $\tilde{L} := LW$, let $P := \tilde{L}^\top(\tilde{L}\tilde{L}^\top)^{-1}\tilde{L}$. Every vector splits as $x = \tilde{L}^\top y + x_0$ with $\tilde{L}x_0 = 0$, a part in the row space of $\tilde{L}$ and a part in its orthogonal complement, the vectors orthogonal to every row. $P$ keeps the first part and removes the second, so $P$ is the orthogonal projection onto the row space of $\tilde{L}$. Since $\Sigma_N L^\top = W\tilde{L}^\top$, $L\Sigma_N L^\top = \tilde{L}\tilde{L}^\top$ and $L = \tilde{L}W^{-1}$, $K^\star L = WPW^{-1}$, and the corrected error is $(I - K^\star L)\varepsilon = W(z - Pz)$. The correction thus subtracts from the whitened noise its orthogonal projection onto the row space of $LW$. It follows that $\phi_{\text{design}} = \operatorname{tr}(K^\star L\Sigma_N)/D = \operatorname{tr}(WPW)/D$. For $\Sigma_N = \sigma^2 I$, $K^\star L = P$ is the orthogonal projection onto $\mathcal{R}(L)$, the projection that the oracle of Section 5.1 removes for each balance. Then $\phi_{\text{design}} = \operatorname{tr} P/n_c = n_r/n_c$, since $\operatorname{tr} P = \operatorname{tr}\left((\tilde{L}\tilde{L}^\top)^{-1}\tilde{L}\tilde{L}^\top\right) = n_r$.

### B.4 Proof of Proposition 2

**Proposition 2 (restated).** *Let Assumption 1 hold, with $\Sigma_\delta$ chosen without using the label, $K^\star$ as in Theorem 1 and $\Sigma_r \succ 0$. For one fixed scene, the fraction of expected squared error that the correction removes, $\phi_{\text{scene}} := (D - \mathbb{E}_\varepsilon\|\hat{c} - c^\star\|^2)/D$, is $\phi_{\text{scene}} = \phi_{\text{design}} - (b - \tau_\delta)/D$. The correction thus improves this scene's label on average over runs if and only if $b - \tau_\delta < D\phi_{\text{design}}$.*

*Proof.* On a fixed scene, $r = \bar{\delta} + L\varepsilon$. Equation (15), with $K = K^\star$ and $\bar{\delta}$ in place of $\delta$, gives $\hat{c} - c^\star = u - K^\star\bar{\delta}$ with $u := (I - K^\star L)\varepsilon$. Here $K^\star$ is fixed, and $\mathbb{E}_\varepsilon u = 0$. Expanding,

$$\|\hat{c} - c^\star\|^2 = \|u\|^2 - 2\,(K^\star\bar{\delta})^\top u + b.$$

The middle term has zero mean. By (10), $\mathbb{E}_\varepsilon\|u\|^2 = \operatorname{tr}\left[(I - K^\star L)\Sigma_N(I - K^\star L)^\top\right]$. Lemma 1 at $K = K^\star$ gives $(I - K^\star L)\Sigma_N(I - K^\star L)^\top = \Sigma_{\text{post}} - K^\star\Sigma_\delta K^{\star\top}$. Taking traces,

$$\mathbb{E}_\varepsilon\|\hat{c} - c^\star\|^2 = \operatorname{tr}\Sigma_{\text{post}} - \tau_\delta + b = D(1 - \phi_{\text{design}}) - \tau_\delta + b,$$

and subtracting it from $D$ and dividing by $D$ gives (6). Without correction, $\mathbb{E}_\varepsilon\|\tilde{c} - c^\star\|^2 = \operatorname{tr}\Sigma_N = D$. The correction lowers this scene's expected squared error exactly when $\phi_{\text{scene}} > 0$, that is, when $b - \tau_\delta < D\phi_{\text{design}}$. On a support, the same steps hold with $\|\cdot\|_M$, $\operatorname{tr}_M$ and $D_M$. □

With $\phi_{\text{net}} = \phi_{\text{design}} - b/D$, (6) also reads $\phi_{\text{scene}} = \phi_{\text{net}} + \tau_\delta/D$. For a gap drawn from the design model, (10) gives $\mathbb{E}_\delta\|K^\star\delta\|^2 = \operatorname{tr}(K^\star\Sigma_\delta K^{\star\top}) = \tau_\delta$, so $b$ averages to $\tau_\delta$ and $\phi_{\text{scene}}$ to $\phi_{\text{design}}$.

**Sample size.** For standard MC, $\Sigma_N = \Sigma_1/N$, and $c^\star$, hence $\bar{\delta}$, does not depend on $N$. Take $\Sigma_\delta = 0$ and $L\Sigma_1 L^\top \succ 0$, so that $\Sigma_r \succ 0$ as required. The factors $1/N$ cancel in $K^\star = \Sigma_1 L^\top(L\Sigma_1 L^\top)^{-1}$, so $K^\star$ and $b$ do not depend on $N$. $\Sigma_{\text{post}}$ and $D$ are both proportional to $1/N$, so neither does $\phi_{\text{design}}$, whereas $b/D = Nb/\operatorname{tr}\Sigma_1$ grows like $N$. Since $LK^\star = L\Sigma_1 L^\top(L\Sigma_1 L^\top)^{-1} = I$, $K^\star\bar{\delta} = 0$ would give $\bar{\delta} = LK^\star\bar{\delta} = 0$. Hence $b > 0$ on the whole field whenever $\bar{\delta} \neq 0$. With $\tau_\delta = 0$, (6) gives $\phi_{\text{scene}} = \phi_{\text{design}} - Nb/\operatorname{tr}\Sigma_1$, negative once $N > \phi_{\text{design}}\operatorname{tr}\Sigma_1/b$ when $b > 0$. On a support, the same holds with $b$, $\phi_{\text{design}}$ and $\operatorname{tr}_M\Sigma_1$ taken on the support.

*Remark* 2 (Trust weights). Section 4 uses $\Sigma_\delta \succ 0$, and the results above say three things about this choice. First, a larger $\Sigma_\delta$ can only lower $\phi_{\text{design}}$. Let $\Sigma_\delta \preceq \Sigma'_\delta$, so that $\Sigma'_r \succeq \Sigma_r \succ 0$, and mark with a prime the quantities built from $\Sigma'_\delta$. For every $K$, $\Gamma'(K) = \Gamma(K) + K(\Sigma'_\delta - \Sigma_\delta)K^\top \succeq \Gamma(K) \succeq \Sigma_{\text{post}}$, by fact (a) and Lemma 1. With $K = K^{\star\prime}$, this gives $\Sigma'_{\text{post}} \succeq \Sigma_{\text{post}}$, so $\phi_{\text{design}}$ falls or stays the same, on the whole field and on every support. In particular, with $\Sigma_N \succ 0$ and linearly independent equations, the deployed $\phi_{\text{design}}$ is at most its value at $\Sigma_\delta = 0$, the value of Proposition 1. Second, on a fixed scene, the expected gain is $\phi_{\text{net}} + \tau_\delta/D$ by Proposition 2. $\operatorname{tr}\Sigma_{\text{post}}$ includes $\tau_\delta$, the squared bias that the design model expects from a random gap. On a fixed scene, the actual squared bias $b$ takes its place. $\phi_{\text{net}} = \phi_{\text{design}} - b/D$ subtracts $b$ but keeps $\tau_\delta$ inside $\operatorname{tr}\Sigma_{\text{post}}$. It thus includes both the actual squared bias $b$ and the modelled one $\tau_\delta$, whereas the error on the scene includes only $b$, and it understates the expected gain by $\tau_\delta/D \geq 0$. Third, Proposition 1(iii) compares two balances with $\Sigma_\delta = 0$. Its proof also applies, with both $\Sigma_r \succ 0$, when the coarse design model sums the fine one, $\delta_{\text{coarse}} = A\delta_{\text{fine}}$, so that $\Sigma_{\delta,\text{coarse}} = A\Sigma_{\delta,\text{fine}}A^\top$, since $\Gamma_{\text{coarse}}(K) = \Gamma_{\text{fine}}(KA)$ still holds. Section 4 builds $\Sigma_\delta$ for cell equations only, so deployed gains are not compared across scales.

### B.5 Proof of Corollary 1, and the sign of one estimate

**Corollary 1 (restated).** *Under the assumptions of Proposition 2,* $\widehat{\phi}_{\text{scene}} := \phi_{\text{design}} - (\widehat{b} - \tau_\delta)/D$ *and* $\widehat{\phi}_{\text{net}} := \phi_{\text{design}} - \widehat{b}/D$*, with* $\widehat{b} := \|\tilde{c} - \hat{c}\|^2 - \tau$*, are unbiased estimates of* $\phi_{\text{scene}}$ *and of* $\phi_{\text{net}}$.

*Proof.* The correction $\tilde{c} - \hat{c} = K^\star r = K^\star\bar{\delta} + K^\star L\varepsilon$ is a fixed part plus a zero-mean part, and

$$\|\tilde{c} - \hat{c}\|^2 = b + 2\,(K^\star\bar{\delta})^\top K^\star L\varepsilon + \|K^\star L\varepsilon\|^2.$$

The middle term has zero mean, and (10) gives $\mathbb{E}_\varepsilon\|K^\star L\varepsilon\|^2 = \operatorname{tr}(K^\star L\Sigma_N L^\top K^{\star\top}) = \tau$. Hence $\mathbb{E}_\varepsilon\|\tilde{c} - \hat{c}\|^2 = b + \tau$ and $\mathbb{E}_\varepsilon\widehat{b} = b$. Since $\phi_{\text{design}}$, $\tau_\delta$ and $D$ are fixed, $\mathbb{E}_\varepsilon\widehat{\phi}_{\text{net}} = \phi_{\text{design}} - b/D = \phi_{\text{net}}$, and $\mathbb{E}_\varepsilon\widehat{\phi}_{\text{scene}} = \phi_{\text{design}} - (b - \tau_\delta)/D = \phi_{\text{scene}}$ by Proposition 2. On a support, the same steps hold with $\|\cdot\|_M$, $\operatorname{tr}_M$ and $D_M$. □

Without the subtraction of $\tau$, $\|\tilde{c} - \hat{c}\|^2$ would overstate $b$ by $\tau$ on average. By (12), $\widehat{\phi}_{\text{scene}} = 2\phi_{\text{design}} - \|\tilde{c} - \hat{c}\|^2/D$, the form given after the corollary. Section 4.2 replaces $\phi_{\text{design}}$ and $\tau$ by averages over random probes. These averages are unbiased, use probes independent of the label and enter both estimates linearly. Under the assumptions of the corollary, the estimates thus stay unbiased over runs and probes. The probes do not remove a bias from an inaccurate $\Sigma_N$ or from correction inputs that depend on the label (Section 4.1).

**The sign of one estimate.** We take the traces as exact here, so the variance below counts the MC noise of the label, not the error of the probes. Both estimates target averages over runs. The gain of one label, $(\|\tilde{c} - c^\star\|^2 - \|\hat{c} - c^\star\|^2)/D$, averages to $\phi_{\text{scene}}$ but varies from run to run, and so do the estimates. Each differs from its target by $-(\widehat{b} - b)/D$, so each has variance $\mathrm{Var}_\varepsilon(\widehat{b})/D^2$. If $\varepsilon$ is Gaussian, this variance has a closed form. With $H := K^{\star\top} K^\star$, $\Omega := L\Sigma_N L^\top$ and $\tilde{c} - \hat{c} = K^\star r$, $\widehat{b} = r^\top H r - \tau$, and $r = \bar{\delta} + y$, with $y := L\varepsilon$ Gaussian of mean zero and covariance $\Omega$. Then

$$r^\top H r = \bar{\delta}^\top H \bar{\delta} + 2\,\bar{\delta}^\top H y + y^\top H y.$$

The linear term has mean zero, so its covariance with the quadratic term is the mean of their product. This mean vanishes, since the product is odd in $y$ and $y$ has the same distribution as $-y$. The variance of $r^\top H r$ is thus the sum of the variances of the two terms. The linear term has variance $4\,\bar{\delta}^\top H\Omega H\bar{\delta}$. For the quadratic term, let $G$ be the symmetric matrix with $G^2 = \Omega$, built from the eigenvalues and eigenvectors of $\Omega$ as $W$ in Remark 1. Take $z \sim \mathcal{N}(0, I)$, a standard Gaussian vector. $Gz$ is Gaussian with mean zero and covariance $G^2 = \Omega$, and a Gaussian vector's distribution is set by its mean and covariance, so $y$ has the distribution of $Gz$. By fact (d), $GHG = \sum_k \nu_k g_k g_k^\top$, so $y^\top H y$ has the distribution of $\sum_k \nu_k w_k^2$ with $w_k := g_k^\top z$. The $w_k$ are jointly Gaussian with $\mathbb{E}[w_j w_k] = g_j^\top g_k$, which is 1 for $j = k$ and 0 otherwise. Jointly Gaussian variables with zero covariances are independent, so the $w_k$ are independent standard normal variables, with $\mathbb{E}w_k^2 = 1$ and $\mathbb{E}w_k^4 = 3$. Hence $\mathrm{Var}(w_k^2) = 3 - 1 = 2$, so $\mathrm{Var}(y^\top H y) = 2\sum_k \nu_k^2 = 2\,\mathrm{tr}\left[(GHG)^2\right] = 2\,\mathrm{tr}\left[(H\Omega)^2\right]$, and

$$\mathrm{Var}_\varepsilon(\widehat{b}) = 2\,\mathrm{tr}\left[(H\Omega)^2\right] + 4\,\bar{\delta}^\top H\Omega H\bar{\delta}. \tag{16}$$

Chebyshev's inequality, $\Pr(|Y - \mathbb{E}Y| \geq t) \leq \mathrm{Var}(Y)/t^2$, then bounds the probability that $\widehat{\phi}_{\text{net}}$ has the wrong sign by $\mathrm{Var}_\varepsilon(\widehat{b})/(D\phi_{\text{net}})^2$ when $\phi_{\text{net}} \neq 0$, and likewise for $\widehat{\phi}_{\text{scene}}$ with $\phi_{\text{scene}}$. This bound controls sign errors only when $D|\phi_{\text{net}}|$ is large compared with the standard deviation of $\widehat{b}$. Otherwise it says little. The bound needs $\bar{\delta}$, which is unknown, so Section 5 checks signs against references instead. On a support, $H = K^{\star\top} M K^\star$ and $D_M$ replaces $D$.

**Keeping only positive estimates.** Let $Y$ be either estimate, and $p = \Pr(Y \geq 0)$ with $0 < p < 1$. Then $\mathbb{E}Y = p\,\mathbb{E}[Y \mid Y \geq 0] + (1-p)\,\mathbb{E}[Y \mid Y < 0]$, with $\mathbb{E}[Y \mid Y < 0] < 0 \leq \mathbb{E}[Y \mid Y \geq 0]$. Hence $\mathbb{E}Y - \mathbb{E}[Y \mid Y \geq 0] = (1-p)\big(\mathbb{E}[Y \mid Y < 0] - \mathbb{E}[Y \mid Y \geq 0]\big) < 0$. The estimates that are kept overstate, on average, the gain they estimate, as stated in Section 3.4.

*Remark* 3 (A misspecified $\Sigma_N$). Suppose the true covariance is $\Sigma_N + \Delta \succeq 0$, with $\Delta$ symmetric, whereas $K^\star$, $\tau$ and $D$ are computed from $\Sigma_N$. The proof of Corollary 1 then gives $\mathbb{E}_\varepsilon\|\tilde{c} - \hat{c}\|^2 = b + \tau + \mathrm{tr}(K^\star L\Delta L^\top K^{\star\top})$. $\widehat{b}$ is thus biased by $\mathrm{tr}(K^\star L\Delta L^\top K^{\star\top})$, a term exactly linear in $\Delta$. The correction itself is less sensitive to a small $\Delta$. Let $\Sigma_r^\Delta := L(\Sigma_N + \Delta)L^\top + \Sigma_\delta \succ 0$, let $K^\Delta := (\Sigma_N + \Delta)L^\top(\Sigma_r^\Delta)^{-1}$ be the best matrix for the true covariance, and let $\Gamma^\Delta$ be the second moment (3) under it. Lemma 1 with the true moments gives

$$\Gamma^\Delta(K^\star) - \Gamma^\Delta(K^\Delta) = (K^\star - K^\Delta)\,\Sigma_r^\Delta\,(K^\star - K^\Delta)^\top.$$

The entries of an inverse are rational functions of the entries of the matrix (Cramer's rule), so $K^\Delta$ depends smoothly on $\Delta$ near $\Delta = 0$, where it equals $K^\star$. Hence $K^\star - K^\Delta$ is of order $\|\Delta\|$, and the trace of the display, the excess expected squared error of $K^\star$ over the best correction for the true covariance, is of order $\|\Delta\|^2$. For a small $\Delta$, the bias of $\widehat{b}$ is thus linear in $\Delta$, whereas this excess design risk is quadratic. The second statement concerns the design model, not one fixed scene.

# C Experimental protocol and additional results

This appendix gives the protocol and the numbers behind Section 5, in the same order, and then two experiments that the main text does not use. Appendix A defines the supports and builds $L$, $\Sigma_N$ and $\Sigma_\delta$ in each setting.

## C.1 Scenes, runs and the measured contrast

Table 1 lists the experiments. Their populations are never pooled.

Table 1: The experiments of this appendix. Budgets are nominal particle counts per run. The stored layer is 0–3 m, and the field above 9 m ends at 800 m. On airport scenes, the correlations of $\Sigma_N$ come from the scene's 300k $\mathrm{NO}_x$ run, which is the label's own run for a 300k $\mathrm{NO}_x$ label. A dash means no correction.

| experiment | section | scenes | runs per scene | correction inputs | scored against | support |
|---|---|---|---|---|---|---|
| flat scenes | 5.1, C.2 | 6 flat, one per class | 8 at 250k | these runs, and 8 runs at each of 500k, 1M, 2M | 12M reference from separate runs | both |
| locality oracle | 5.1, C.2 | 3 flat (I, III/1, IV) | 3 at 2M | – | 12M reference from separate runs | above 9 m |
| airport scenes | 5.2, C.3 | 18 small airports, two pollutants | 8 at 3M | these runs, and correlations from a 300k run | mean of the other 7 runs | both |
| control without reuse | 5.2, C.3 | the same 18 | $3 + 5$ at 3M | the first 3 runs | mean of the other 4 scored runs | both |
| deployment inputs | 5.2, C.3 | the same 18 | 1 at 300k | one other 3M run, and correlations from a 300k run | mean of the 7 remaining runs | stored layer |
| held-out inputs | C.4 | the same 18 | 1 at 3M | other runs | the scene's eight-run contrast | above 9 m |
| training labels | C.4 | 5,995 $\mathrm{NO}_x$, 5,992 PM-1 | 1 at 300k, 1 at 3M | the 3M run, and correlations from a 300k run | none | above 9 m |
| noise dominance | 5.3, C.5 | 2,000 training, 600 validation | 300k or 3M labels | – | 3M labels | 0–3 m layer |
| direct solve | C.6 | 6 flat | – | – | 12M reference from separate runs | above 9 m |
| weighted losses | C.6 | 5,995 training, 3,600 validation, $2 \times 600$ test | 300k labels | – | 3M labels | 0–3 m layer |

**Flat scenes.** Each stability class has one flat scene, built outside AirDispNet, with no building and a single point source that does not deposit. The wind and the turbulence vary only with height, following the simulator's boundary-layer model for the class. The wind blows along the grid's $x$ axis near the ground and turns with height. Each scene is run eight times at each of 250k, 500k, 1M and 2M particles, with seeds 0 to 7. The 250k runs are the scored labels. All four budgets enter $\Sigma_\delta$, and the 2M runs also give the correlations of $\Sigma_N$. The reference is the mean of two further runs of 6M particles with other seeds, 12M particles in all, so it shares no particle with any label. The locality oracle uses three 2M runs of classes I, III/1 and IV from a separate set of runs of the same scenes, with a 12M reference built in the same way.

**Airport scenes.** The replicate scenes are 18 AirDispNet scenes, three per stability class, all of the small-airport category, with one to three buildings on a square domain 700 m wide. Each is run eight times at 3M particles with independent seeds, for $\mathrm{NO}_x$ and PM-1 together. This joint run placed the PM-1 sources elsewhere than in AirDispNet's stored field of the same scene, with the same geometry, weather and emission budget. All PM-1 results on the replicate scenes use these source positions. $\Sigma_\delta$, the variances of $\Sigma_N$ and the scored cells come from the eight runs, and the correlations of $\Sigma_N$ from the scene's separate 300k $\mathrm{NO}_x$ run. Each run is scored against the mean of the other seven.

**Training labels.** The deployment corrects the 300k labels of 5,995 $\mathrm{NO}_x$ and 5,992 PM-1 training scenes, each with its own full-budget run, as in Section 4. No reference exists for these scenes.

Table 2: Gains by stability class, from very stable (I) to convective (V). Flat rows give the median over the eight 250k runs of one scene, scored against the 12M reference. Airport rows give the median over the three scenes of a class of each scene's median over its runs. On the stored layer, $NO_x$ class I and PM-1 class II have two scored scenes only. Airport contrasts at 3M are plug-in quantities. The 300k rows use one label per scene, corrected with the variances and $\Sigma_\delta$ of one other 3M run and the correlations of the scene's 300k $NO_x$ run, and scored against the mean of the seven remaining runs. $\widehat{\phi}_{\mathrm{net}}$ and $\widehat{\phi}_{\mathrm{scene}}$ are the estimates of Corollary 1, averaged over a scene's runs and computed on the support of their row. The last row of each pollutant is the share of its 300k training labels that pass the screen computed above 9 m, a pass rate and not a measured gain.

| | | I | II | III/1 | III/2 | IV | V |
|---|---|---|---|---|---|---|---|
| flat, 0–3 m | $\phi_{\mathrm{design}}$, predicted | 0.827 | 0.856 | 0.827 | 0.825 | 0.836 | 0.840 |
| | $\phi_{\mathrm{meas}}$, measured | 0.680 | 0.768 | 0.782 | 0.787 | 0.796 | 0.771 |
| flat, >9 m | $\phi_{\mathrm{design}}$, predicted | 0.771 | 0.877 | 0.812 | 0.844 | 0.861 | 0.869 |
| | $\phi_{\mathrm{meas}}$, measured | 0.555 | 0.815 | 0.760 | 0.799 | 0.826 | 0.841 |
| $NO_x$ | $\phi_{\mathrm{meas}}$, 0–3 m | −0.54 | −0.49 | −0.03 | +0.52 | +0.43 | +0.59 |
| | $\widehat{\phi}_{\mathrm{net}}$, 0–3 m | −0.40 | −0.46 | −0.08 | +0.43 | +0.31 | +0.47 |
| | $\phi_{\mathrm{meas}}$, 0–3 m, 300k | +0.30 | +0.25 | +0.39 | +0.54 | +0.66 | +0.74 |
| | $\widehat{\phi}_{\mathrm{net}}$, 0–3 m, 300k | +0.30 | +0.49 | +0.47 | +0.81 | +0.60 | +0.74 |
| | $\phi_{\mathrm{meas}}$, >9 m | −4.18 | −0.91 | −0.84 | −1.21 | −0.18 | +0.65 |
| | noise-only gain, >9 m | 0.14 | 0.44 | 0.11 | 0.44 | 0.62 | 0.82 |
| | $\widehat{\phi}_{\mathrm{net}}$, >9 m | −4.32 | −1.05 | −0.27 | −0.58 | −0.07 | +0.55 |
| | $\widehat{\phi}_{\mathrm{scene}}$, >9 m | −4.26 | −0.97 | −0.22 | −0.47 | −0.03 | +0.61 |
| | passing the screen at 300k (%) | 19.2 | 60.5 | 68.2 | 71.3 | 93.1 | 98.3 |
| PM-1 | $\phi_{\mathrm{meas}}$, 0–3 m | −0.61 | −0.86 | −0.02 | +0.50 | +0.36 | +0.65 |
| | $\widehat{\phi}_{\mathrm{net}}$, 0–3 m | −0.26 | −0.72 | −0.09 | +0.37 | +0.24 | +0.51 |
| | $\phi_{\mathrm{meas}}$, 0–3 m, 300k | +0.48 | −0.08 | +0.50 | +0.43 | +0.50 | +0.70 |
| | $\widehat{\phi}_{\mathrm{net}}$, 0–3 m, 300k | +0.51 | +0.61 | +0.32 | +0.65 | +0.65 | +0.77 |
| | $\phi_{\mathrm{meas}}$, >9 m | −6.21 | −2.88 | −1.39 | −1.06 | −0.69 | +0.53 |
| | noise-only gain, >9 m | 0.16 | 0.33 | 0.06 | 0.16 | 0.42 | 0.76 |
| | $\widehat{\phi}_{\mathrm{net}}$, >9 m | −8.13 | −2.54 | −1.05 | −0.61 | −0.53 | +0.42 |
| | $\widehat{\phi}_{\mathrm{scene}}$, >9 m | −8.01 | −2.48 | −0.99 | −0.54 | −0.47 | +0.49 |
| | passing the screen at 300k (%) | 23.8 | 51.6 | 54.5 | 56.6 | 79.6 | 91.9 |

**The measured contrast.** Equation (8) sums over one of the two supports of Section 4.2. A run counts only if the denominator is positive, and a scene reports the median over its runs. On flat scenes, $v_{\mathrm{ref}}$ is the sample variance of the eight 250k runs, summed over the support and scaled by the $1/N$ law to the 12M particles of the reference. On airport scenes, it is the summed sample variance of the eight runs divided by seven, the variance of a mean of seven runs. Let $M$ be the diagonal matrix with ones on the support and zeros elsewhere. If the eight runs are independent, with the same mean and covariance $\Sigma_N$, and the support is fixed, then $\mathbb{E}\|\tilde{c} - c_{\mathrm{ref}}\|_M^2 = \frac{8}{7}\,\mathrm{tr}(M\Sigma_N)$ and $\mathbb{E}v_{\mathrm{ref}} = \frac{1}{7}\,\mathrm{tr}(M\Sigma_N)$. The denominator thus averages to $D_M = \mathrm{tr}(M\Sigma_N)$, the scale of the screen, and the same holds on flat scenes.

The label shares no particle with its reference, but its correction can depend on the reference runs. Write the correction as $\hat{c} = \tilde{c} - h$ and the reference as $c_{\mathrm{ref}} = c^\star + \eta$, with $\eta$ its MC error. Expanding both squares, the numerator of (8) equals $\|\tilde{c} - c^\star\|_M^2 - \|\hat{c} - c^\star\|_M^2 - 2\eta^\top M h$, the reduction of the squared error from $c^\star$ minus a cross term. For a fixed support, if $h$ is built without the reference runs, $\eta$ has zero mean and is independent of $h$, so the cross term averages to zero. This concerns the numerator only, and a ratio of two unbiased terms is not itself unbiased. On flat scenes, the reference enters the correction only through thresholds, which select the cells where the correlations are fitted and the cells that set the floor $\gamma$ of $\Sigma_\delta$. The same threshold selects the support. Taking these thresholds on the mean of the eight 2M runs instead changes no median contrast by more than $0.001$. On airport scenes, $h$ and the support use all eight runs, including those that form the reference, so the cross term need not vanish. The contrast there is a plug-in quantity, computed with inputs estimated from the runs it scores, and not an unbiased estimate of the gain.

**The noise-only gain.** Above 9 m, a plug-in control applies the same correction with $q$ replaced by $q + \hat{\bar{\delta}}$, where $\hat{\bar{\delta}}$, the mean residual of the scene's eight runs, estimates the closure gap. If $\hat{\bar{\delta}}$ equalled $\bar{\delta}$, $b$ would vanish, and by Proposition 2 this gain would approach $\phi_{\mathrm{design}} + \tau_\delta / D$ rather than $\phi_{\mathrm{scene}}$. The estimate also contains the MC error of the eight-run mean, so this reading is an idealization. The control reuses the eight runs, as the contrast does.

### C.2 Flat scenes

**Gains.** All 48 runs improve on both supports, the smallest contrast being 0.66 on the stored layer and 0.51 above 9 m. On the stored layer, $\phi_{\mathrm{design}}$ exceeds the median contrast by 0.038 to 0.088 in five classes and by 0.147 in class I. Above 9 m, it exceeds it by 0.028 to 0.062, and by 0.216 in class I. Since $c^\star$ is the mean of the raw label, the raw label's expected squared error is its MC variance, and $1 - \phi_{\mathrm{meas}}$ estimates the share of it that the corrected label keeps. This share is 20 to 32% on the stored layer and 16 to 44% above 9 m. These data cannot tell a small error in $\Sigma_N$ from a small bias. The relative error $\|c - c_{\mathrm{ref}}\|_M / \|c_{\mathrm{ref}}\|_M$, with $c$ the label before or after correction, falls from 5.0–6.4% to 2.8–3.0% on the stored layer. Every stored-layer solve reaches a relative residual of $10^{-6}$ within 447 iterations.

**Reuse of the scored run.** $\Sigma_\delta$ and the variances of $\Sigma_N$ both use the scored run, so its own error shapes its correction. Leaving it out of the variances of $\Sigma_N$ gives a median contrast of 0.62 to 0.72 on the stored layer and 0.49 to 0.81 above 9 m. As a precaution, this variant also leaves out of the correlation fit the 2M run that has the scored run's seed, since runs of equal seed start from the same random state. $\Sigma_\delta$ still uses the scored run.

**The noise model.** In a separate set of runs of classes I, III/1 and IV, the error that one run reports predicts the standard deviation between runs on the stored layer with median ratios of 1.004, 1.002 and 1.027. The correlations of $\Sigma_N$ act through the residual $L\tilde{c}$. We sum over the field above 9 m the variance of the residual that the model of Appendix A predicts, and the variance measured across runs. The square root of the first sum over the second, a ratio of standard deviations, is the calibration ratio. A ratio of one means that the predicted and measured sums of residual variances agree. Across the eight 250k runs, it is 0.995 to 1.043 in the six classes. In the separate runs, a diagonal $\Sigma_N$, with no correlation between cells, gives 1.64, 1.23 and 1.13 in classes I, III/1 and IV. Neighbouring errors are positively correlated, so the differences in $L$ partly cancel them. Fitting the correlations layer by layer matters most in the very stable class. There, a single kernel for all layers, measured at the surface for the class, over-predicts the standard deviation by 22% and gives a gain of 0.483 above 9 m, against 0.555 with the fit layer by layer. Both values use variances that include the scored run. Leaving the 2M run with the scored run's seed out of the layer-by-layer fit gives 0.552.

**Generic smoothers.** Two denoisers act on the label alone. A Gaussian filter averages each cell with its neighbours, with weights that decay with distance, and total-variation denoising smooths while keeping sharp edges. Each receives, in every class, the parameter that brings the runs closest to the 12M reference. The correction uses the reference only through the thresholds of Appendix C.1, whereas the smoothers are tuned to minimize their error against it. On the stored layer, the median relative error over the eight runs is then 3.2–4.0% with the Gaussian filter and 3.9–4.2% with total variation, against 2.8–3.0% for the correction and 5.0–6.4% for the raw label.

**The locality oracle.** For a 2M run $\tilde{c}$ and its 12M reference $c_{\mathrm{ref}}$, the difference $e = \tilde{c} - c_{\mathrm{ref}}$ is mostly the run's noise. A coarsening matrix $G$ sums groups of cell equations, which gives the coarse balance $A = GL$. The reference satisfies it exactly, with right-hand side $Ac_{\mathrm{ref}}$. Let $P$ be the orthogonal projector onto the row space of $A$, over the whole grid with equal weights on all cells. Since $e - Pe$ lies in the null space of $A$, $APe = Ae$, so $Pe$ is the part of $e$ that the coarse balance measures. Any solution $y$ of $AA^\top y = Ae$ gives $Pe = A^\top y$. We approximate $Pe$ by $p_{250} = A^\top y_{250}$, where $y_{250}$ is the iterate after at most 250 conjugate-gradient iterations on this system. We report $1 - \|e - p_{250}\|_M^2 / \|e\|_M^2$, the reduction of the squared gap between run and reference on $M$, averaged over three runs, with $M$ the field above 9 m without a small box around the source. As $M$ is not the whole grid, this reduction is not exactly the share of $\|e\|_M^2$ carried by the projection. Table 3 gives every scale. Running 5,000 iterations instead changes the block entries by less than 0.01 percentage points. It raises the cell entries to 99.38, 99.93 and 99.95, with the cell

equations still solved only to a maximum relative residual of $3 \times 10^{-3}$. This oracle uses the reference and weights all cells equally, so it shows where the noise lies, not the gain of the correction.

Table 3: Reduction (%) of the squared gap between a 2M run and its 12M reference when its approximate projection onto the row space of a balance is removed, by scale, on the flat scenes of three stability classes (mean of three runs, field above 9 m). Blocks are square groups of cells within one layer. Each solve stops after at most 250 iterations.

| balance equations | number | I | III/1 | IV |
|---|---|---|---|---|
| one for the domain | 1 | 0.00 | 0.00 | 0.00 |
| one per layer | 42 | 0.007 | 0.005 | 0.020 |
| blocks of $20 \times 20$ cells | 756 | 0.69 | 0.53 | 0.62 |
| blocks of $10 \times 10$ cells | 2,520 | 2.41 | 2.21 | 2.41 |
| blocks of $5 \times 5$ cells | 10,080 | 11.66 | 9.83 | 8.47 |
| one per cell | 252,000 | 99.25 | 99.68 | 99.88 |

### C.3 AIRPORT SCENES

**The stored layer at 3M.** In one scene per pollutant, $\mathrm{NO}_x$ scene 24 (class I) and PM-1 scene 1 (class II), no cell of the stored layer clears the concentration threshold, because the plume does not reach the ground in appreciable concentration. These two scenes are not scored, which leaves 17 per pollutant. Scored on the whole lowest layer instead, they give $-1.54$ and $-0.37$. The correction improves 10 of the 17 scenes for each pollutant, with the class medians of Table 2.

**A control without reuse.** Here $\Sigma_\delta$, the variances of $\Sigma_N$ and the scored cells come from the first three runs only. Each of the other five runs is scored against the mean of the remaining four, so no run that builds the correction is corrected or serves as a reference. On the stored layer, the correction then improves 8 of 17 $\mathrm{NO}_x$ and 9 of 18 PM-1 scenes. The stored-layer support, now taken from the three runs, is empty only for $\mathrm{NO}_x$ scene 24. The class medians keep the signs of the main protocol, at $-1.18$, $-1.42$, $-0.26$, $+0.19$, $+0.25$ and $+0.48$ for $\mathrm{NO}_x$ and $-1.89$, $-0.65$, $-0.31$, $+0.35$, $+0.32$ and $+0.52$ for PM-1, from class I to V. Above 9 m, they range from $+0.51$ and $+0.36$ in the convective class to $-6.78$ and $-27.0$ in the most stable one, below the main protocol. This control changes inputs and scoring together, so the drop cannot be attributed to removing the reuse alone.

**Above 9 m.** The contrast of the same 3M corrections is negative for 14 of 18 $\mathrm{NO}_x$ and 16 of 18 PM-1 scenes, whereas the noise-only gain is positive for 33 of 36 scene–pollutant pairs (Figure 3a). Removing noise thus does not imply an improvement on this support. The stored layer is noisier than this field. The median relative error of the raw label against its reference is $2.0\%$ there against $0.9\%$ above 9 m for $\mathrm{NO}_x$, and $2.8\%$ against $1.4\%$ for PM-1.

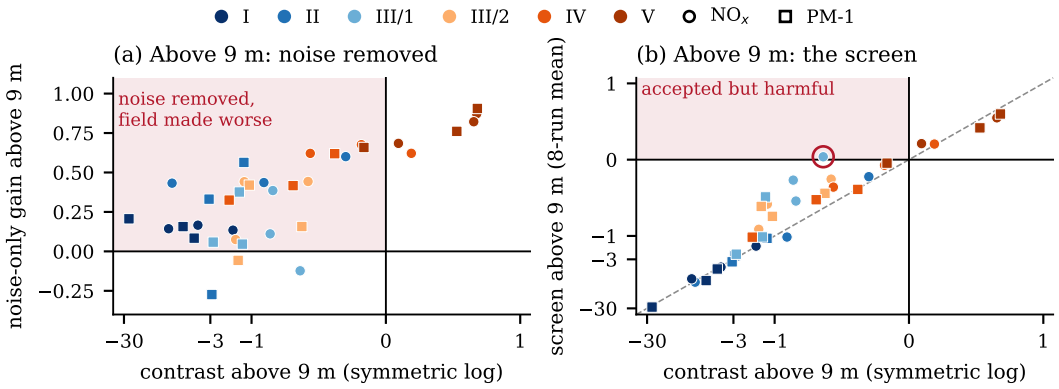


Figure 3: The 3M corrections of Figure 2, scored on the field above 9 m, for 18 scenes per pollutant, each summarized over its eight runs. (a) Noise-only gain against the contrast. The shaded region holds pairs whose noise-only gain is positive but whose contrast is negative. (b) The screen $\widehat{\phi}_{\text{net}}$ against the same contrast. The shaded region holds harmful corrections that the rule would accept, and the ring marks the only one, for $\text{NO}_x$.

**At the training budget.** Here every input is taken as the deployment takes it, following a plan fixed before the run. The label is a 300k run. The variances of $\Sigma_N$, $\Sigma_\delta$ and the scored cells come from one other 3M run, and the correlations from the scene's 300k $\text{NO}_x$ run, as in Section 4.1. The label, raw and corrected, is scored on the stored layer against the mean of the seven remaining runs, which share no particle with it or with its inputs, and $v_{\text{ref}}$ is their summed sample variance divided by seven. For $\text{NO}_x$, the label is the one the deployment stored, which the rerun reproduces to a relative $2 \times 10^{-6}$. For PM-1, whose stored labels belong to other source positions, we simulated a new 300k run of each replicate scene with the transport code of the replicate runs. The same two scenes as at 3M have an empty stored layer. The correction improves 17 of 17 $\text{NO}_x$ and 14 of 17 PM-1 labels, against 10 of 17 for each pollutant at 3M (Figure 4a). For every raw label, $\|\tilde{c} - c_{\text{ref}}\|_M^2 - v_{\text{ref}}$ is 0.87 to 1.18 times $D_M$, the variance that $\Sigma_N$ predicts. This checks the variances of $\Sigma_N$, not its correlations. For $\text{NO}_x$, the correlations fitted as in deployment on the scene's 300k run, combined with the variances measured across the eight 3M runs, give a calibration ratio above 9 m of 0.88 to 1.11 over the 18 scenes, with a median of 1.01. In 600 $\text{NO}_x$ deployment scenes drawn at random, 100 per class, the along-wind decay length $\lambda_{\parallel,z}$ of Appendix A sits at the upper end of the searched range, 12 cells, in 45 to 47% of the fitted layers in classes I, II and III/1, and in 25 to 35% in the three others. In those layers, the best fit may lie beyond the range.

### C.4 The screen

A correction passes the screen when $\widehat{\phi}_{\text{net}} \geq 0$ (Section 4.2). No label is ever discarded, so "accept" and "reject" name what this rule would decide. The error of the screen is its estimate minus the contrast it is compared with. Unless stated otherwise, that contrast is taken on the support where the screen is computed. In the eight-run experiments, a single run's screen is compared with its scene's contrast, the median over the scene's runs, and not with the run's own.

**The stored layer at 3M.** We computed the screen on the stored layer of every scored scene, following a plan fixed before the run. The plan kept the rule and set a benchmark of at least 15 of 17 scene signs right with no class median wrong. Averaged over a scene's eight runs, the screen has the sign of the contrast for 16 of 17 $\text{NO}_x$ and all 17 PM-1 scenes, with median absolute errors of 0.10 and 0.11, and every class median has the right sign. The benchmark is thus met for both pollutants. The screen accepts one harmful correction, $\text{NO}_x$ scene 49 (class II) with estimate $+0.18$ and contrast $-0.15$, and rejects none that helps. Always predicting improvement would be right for 20 of the 34 scene–pollutant pairs. From a single run, the screen has the sign of its scene's contrast for 128 of 136 $\text{NO}_x$ and 133 of 136 PM-1 estimates, eight runs in each of 17 scenes, clustered within scenes rather than independent. Its median signed errors are $-0.07$ and $-0.08$ and its median

absolute error 0.12 for both, so it slightly understates the gain. Without the noise term $\tau$, that is with $\widehat{b}$ replaced by $\|\tilde{c} - \hat{c}\|^2$, it is right for only 7 of 17 scenes for each pollutant.

**Deployment inputs at 300k.** With the inputs of Appendix C.3, following a plan fixed before the run, the stored-layer screen has the sign of the contrast for 16 of 17 $NO_x$ and 15 of 17 PM-1 labels (Figure 4b). These 31 of 34 equal the count of always predicting improvement. The screen rejects one $NO_x$ correction that helps, scene 2 with estimate $-0.03$ and contrast $+0.15$, and accepts two PM-1 corrections that hurt, scene 24 (class I, $+0.47$ against $-0.58$) and scene 49 (class II, $+0.62$ against $-0.39$). It is optimistic, more so for PM-1, with median signed errors of $+0.04$ and $+0.16$ and median absolute errors of 0.10 and 0.16. Every $NO_x$ class median has the right sign, and every PM-1 one except class II, whose two scored scenes give $+0.61$ against $-0.08$. The benchmark fixed in advance, at most two wrong signs and no class median wrong, is thus met for $NO_x$ and not for PM-1. Without $\tau$, only 0 and 3 of 17 signs are right.

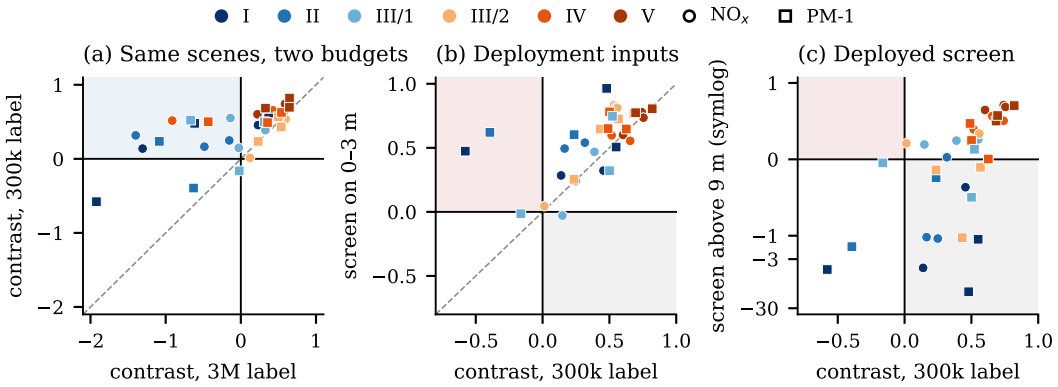


Figure 4: The screen with deployment inputs. Each point is one 300k label of a replicate scene, 17 per pollutant, corrected with the variances and $\Sigma_\delta$ of one other 3M run and the correlations of the scene's 300k $NO_x$ run, and scored on the stored layer against the mean of the seven remaining runs. (a) Contrast of the 300k label against the plug-in contrast of the same scene at 3M (Figure 2). The shaded region holds scenes that are worse at 3M but better at 300k. (b) The screen computed on the stored layer. (c) The screen computed above 9 m, whose pass rates are reported for the training labels. In (b) and (c), the shaded regions hold positive screens with harmful corrections and negative screens with helpful ones.

**Above 9 m.** With the eight-run inputs, the screen averaged over a scene's runs has the sign of the contrast above 9 m for 17 of 18 $NO_x$ and all 18 PM-1 scenes, and of every class median, with median absolute errors of 0.174 and 0.291 (Figure 3b). From a single run, it is right for 137 of 144 $NO_x$ and 144 of 144 PM-1 estimates, with median signed errors of $+0.124$ and $+0.122$ and median absolute errors of 0.186 and 0.276. Unlike the stored-layer screen at 3M, it overstates the gain. Without $\tau$, it is right for 14 and 16 of 18 scenes. It accepts one harmful $NO_x$ correction, a class III/1 scene with contrast $-0.64$ and estimate $+0.04$, and none for PM-1. The estimate $\widehat{\phi}_{\text{scene}}$ adds $\tau_\delta/D$, estimated at 0.039 to 0.112 (median 0.064) for $NO_x$ and 0.050 to 0.122 (median 0.067) for PM-1. It has the right sign for 17 of 18 scenes for both pollutants and for 133 of 144 and 139 of 144 single runs, with median signed errors of $+0.183$ for both and median absolute errors of 0.193 and 0.285.

**From one support to the other.** Read against the contrast of the stored layer, the screen computed above 9 m is right for only 8 of 17 $NO_x$ and 9 of 17 PM-1 scenes at 3M. It rejects 7 $NO_x$ and 8 PM-1 corrections that improve the stored label, and accepts two $NO_x$ corrections that worsen it, scene 4 (class IV, contrast $-0.91$) and scene 2 (class III/1, $-0.03$). With deployment inputs at 300k, it is right for 13 and 10 of 17 labels. It then rejects 4 of the 17 $NO_x$ and 7 of the 14 PM-1 corrections that improve the stored label, and accepts none that worsens it (Figure 4c).

**Held-out inputs at 3M.** A stress test fixed in advance scores, above 9 m, the one-run screen of a 3M run whose inputs all come from other runs. The variances of $\Sigma_N$ come from a second run, the correlations from the scene's 300k run and $\Sigma_\delta$ from $c_{\mathrm{ref}}$, the mean of the seven other runs. The first rule asks the screen to have the sign of the scene's eight-run contrast for at least 16 of 18 scenes, with no class median wrong. The second compares the corrected label $\hat{c}_{\mathrm{held}}$ with $\hat{c}_8$, the correction of the same run $\tilde{c}$ with $\Sigma_\delta$ taken from all eight runs, through $d = \|\hat{c}_{\mathrm{held}} - \hat{c}_8\|_M / \|\tilde{c} - c_{\mathrm{ref}}\|_M$ on the field above 9 m. It requires a median $d \le 0.25$ over scenes and $d \le 0.50$ in every scene. $\mathrm{NO}_x$ fails both, with 15 of 18 signs, 5 of 6 class medians, a median $d$ of 0.28 and a largest $d$ of 0.54. PM-1 passes the first, with 17 of 18 and 6 of 6, and fails the second, with 0.26 and 1.12.

**Pass rates on the training labels.** At 300k, 68.5% of the $\mathrm{NO}_x$ and 59.7% of the PM-1 training labels pass the screen computed above 9 m, from 19% in class I to 98% in class V (Table 2). No reference exists for these scenes to tell which passing corrections improve their label, so these are pass rates, not measured gains. PM-1 passes less often in five of the six classes, and slightly more in class I, 23.8% against 19.2%.

### C.5 The learner

**Architecture and training.** The learner of Section 5.3 is a factorized Fourier neural operator with 37.2M parameters, trained on $\mathrm{NO}_x$ labels only. A vertical encoder, with two blocks, 32 channels and 12 vertical Fourier modes, reduces 17 three-dimensional input fields on the 33 lowest levels to a 96-channel horizontal field, joined with 12 two-dimensional fields. Four factorized spectral blocks with 96 channels then retain up to 250 Fourier modes per horizontal axis, fewer on grids too small to provide them. This count comes from a cutoff of 0.0998 cycles per metre on the 500-cell reference grid with 5 m cells. Twenty scalars, for the weather, the stability class and deposition, enter through feature-wise affine conditioning. The output is the 0–3 m layer, divided by the pollutant's emission rate and multiplied by the 10 m wind speed, floored at $0.7\,\mathrm{m\,s^{-1}}$, and by $(100\,\mathrm{m})^2$. Unless stated otherwise, training minimizes the mean squared error over the domain outside building footprints with AdamW, a learning rate of $3 \times 10^{-4}$, a weight decay of $10^{-4}$, 200 linear warm-up steps and a cosine decay whose horizon counts micro-batches, so that the rate ends near $2.6 \times 10^{-4}$. Gradients are clipped at norm 1 and accumulated over four micro-batches of one scene, four on the smallest grids, in float16 mixed precision. A run of 40 epochs on 2,000 scenes took about 19 h on one NVIDIA V100 GPU.

**The noise-dominance test.** Section 5.3 trains this operator for 40 epochs on 2,000 training scenes, with seeds 0, 1 and 2, once with 300k labels and once with 3M labels. Each 300k label uses one of the ten particle groups of its 3M run, so the two arms share particles. Both arms are scored with their final weights on 600 validation scenes against the 3M labels, the test split staying locked. A scene's squared error is averaged, in $(\mathrm{g\,m^{-3}})^2$, over the cells of the 0–3 m layer outside buildings where the label or the prediction exceeds $10^{-6}\,\mathrm{g\,m^{-3}}$. The statistic fixed in advance is the median over scenes of the paired difference in squared error, 300k arm minus 3M arm, then the median over seeds. It is $-3.9 \times 10^{-13}$, with $-1.9 \times 10^{-11}$, $6.7 \times 10^{-11}$ and $-3.9 \times 10^{-13}$ at the three seeds. The yardstick is the standard deviation over seeds of an arm's mean squared error, the larger of the two arms, $5.9 \times 10^{-8}$. The criterion required the statistic to reach three times this spread, whereas it is $-6.6 \times 10^{-6}$ times the spread (Figure 5b). The arms' mean errors are 4.08, 5.22 and $4.92 \times 10^{-7}$ at 300k and 4.32, 4.17 and $4.37 \times 10^{-7}$ at 3M, so their difference changes sign across seeds.

**The operator's error against the MC floor.** Section 5.3 compares the typical error of another operator with the floor, the mean over the scored cells of the 3M label's MC variance. This operator has the same architecture and adds a source anchor $y_{\mathrm{src}}$, a sum over the sources of closed-form Gaussian plumes, whose cross-section is a Gaussian that widens downwind. The anchor is built from the source list, the scene's wind and turbulence profile and the pollutant's emission rate, and reads no label. The operator predicts $g(x)\, y_{\mathrm{src}} + \mathrm{net}(x)$, where $x$ denotes the scene's input fields, net the network and $g$ a learned multiplicative field, equal to one at the start of training. It was trained with one seed on 2,000 scenes for 25 epochs. On 600 validation scenes, the median over scenes of the ratio of its squared error to the floor is 31, the value of Section 5.3. It is 71 for the additive variant $y_{\mathrm{src}} + \mathrm{net}(x)$. The ratio of the mean error to the mean floor is far larger, about 497 and 504, because a few scenes with sharp peaks near the sources dominate both means (Figure 5c). The

non-dimensional units of training rescale each scene by its own factor, which changes the weight of each scene in the two means but not the ratio of any one scene. Figure 5a shows this concentration for an operator of the same architecture trained with a mean-squared-error loss on 300k labels. On 200 validation scenes, the top $0.1\%$ of its cells carry $98\%$ of its squared error. In the training's non-dimensional units, its error is $1{,}986$ times the floor, and $83$ times without the top $1\%$ of cells.

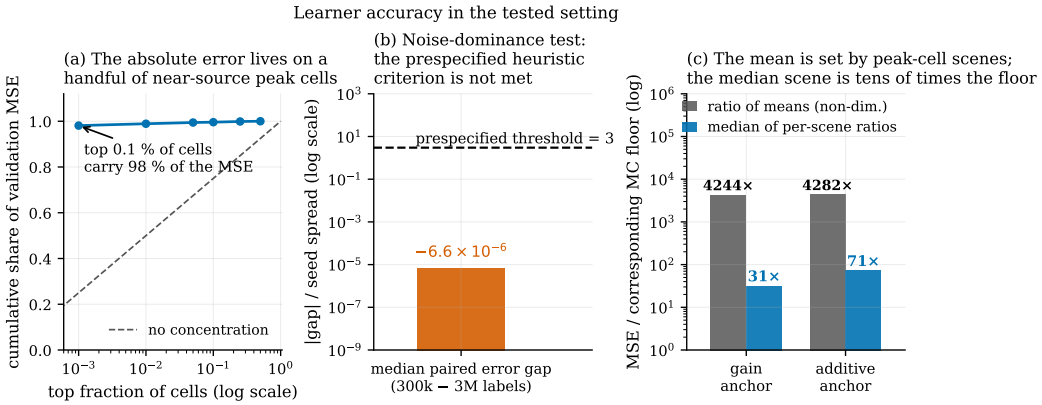


Figure 5: The learner. (a) Share of the squared error carried by the cells of largest error, for an operator trained with a mean-squared-error loss on 300k labels (200 validation scenes). The top $0.1\%$ of cells carry $98\%$ of it. (b) Noise-dominance test. The paired error gap between the operators trained on 300k and 3M labels (2,000 scenes, three seeds, $40$ epochs), divided by the seed-to-seed spread, against the prespecified threshold of three. (c) Two anchored operators (600 validation scenes, one seed, $25$ epochs). Ratio of the mean squared error to the mean MC floor, in the training's non-dimensional units, next to the median over scenes of the per-scene ratio, which does not depend on units. In physical units, the first ratio is about $497$ and $504$.

### C.6 Two further experiments

**Solving the balance directly.** If the balance were exact and had a unique solution, solving it would give the target without any particle. For the centred $L$ of Appendix A, a restarted Krylov solver, an iterative method for large linear systems, stalls at a relative residual of $9 \times 10^{-2}$ after $3{,}000$ iterations. We therefore discretize the same transport equation in flux-conservative form, with the Scharfetter–Gummel scheme, a standard scheme for transport and diffusion that stays stable when the wind dominates. It keeps the same wind and diffusivity except the cross term $\kappa_{xy}$. The concentration is set to zero on the inflow, lateral and top faces, the outflow face lets the plume leave with the wind, and no flux crosses the ground. The system is solved directly to a relative residual of $10^{-15}$. Above $9$ m on the six flat scenes, the relative error of this solution against the 12M reference is $0.300$, $0.401$, $0.630$, $0.626$, $0.617$ and $0.598$ from class I to V, $14$ to $21$ times that of a 250k label ($0.0145$ to $0.0426$). With this discretization and these boundary conditions, the direct solution is thus far less accurate than a 250k label.

**Weighting the loss instead of correcting the label.** The covariance can weight the training loss instead, leaving the labels unchanged. We trained the operator of Appendix C.5 on $5{,}995$ scenes with 300k labels for $40$ epochs, at three seeds, with three losses. Plain MSE weights all cells equally. Inverse-variance weighting divides each cell's squared error by its MC variance. Generalized least squares (GLS) uses the whole $\Sigma_N$, correlations included. A scene's score is the relative error $\|\hat{y} - y\|/\|y\|$ of the prediction $\hat{y}$ against the 3M label $y$, over all cells of the stored layer outside buildings, with no concentration threshold, first taken as the median over the three seeds. Arms are compared scene by scene. On the $3{,}600$ validation scenes, GLS improves on MSE by a paired median of $0.052$, with arm medians of $0.202$ and $0.271$, and is better on $83\%$ of scenes. Inverse-variance weighting leaves the paired median at zero. Out of distribution, the order reverses. Two folds, that is two splits of the data, each remove one stable class, I or II, from training and validation, and score the $600$ test scenes of that class. MSE is then better on the typical held-out scene of both folds, by $0.033$ to

0.047, and by 0.017 to 0.070 at each seed separately. For the four comparisons with MSE, two-sided Wilcoxon signed-rank tests reject, with $p$ from $1.6 \times 10^{-66}$ to $1.1 \times 10^{-21}$, the hypothesis that the paired differences are distributed symmetrically about zero. All four survive a Holm correction, which bounds the chance of any false positive among the six comparisons of the two folds. The twelve per-seed comparisons with MSE survive a Holm correction over the 18 per-seed ones. The weighted losses are still better on 15 to 25% of held-out scenes, those where MSE fails worst. On fold I, these few scenes reverse the mean paired difference, $-0.168$ for GLS and $-0.127$ for inverse variance, since the MSE arm reaches a relative error of 48.5 on one scene, against at most 1.26 and 4.93 for the weighted arms. All three arms are poor on the held-out class, with median relative errors of 0.72 to 0.79 on fold I and 0.46 to 0.50 on fold II. These experiments test loss weighting, not training on corrected labels.

# D THE AIRDISPNET DATASET

## D.1 CORPUS AND FROZEN SPLITS

AirDispNet contains 24,000 synthetic scenes, 1,000 in each of the 24 combinations of four airport categories and six Klug–Manier stability classes. The four procedural categories are a small airport, a medium airport, a large hub and a cargo airport. A scene fixes the airport geometry, the emission inventory and the sampled weather. Inventories follow published aviation methods (International Civil Aviation Organization, 2024; 2017; 2020; Celikel et al., 2005; Janicke et al., 2007).

Each scene has two pollutants, $\mathrm{NO}_x$ and particulate matter $\mathrm{PM}_{10}$. The stored $\mathrm{PM}_{10}$ is the sum of two AUSTAL components transported separately, `pm-1` and `pm-2`, written PM-1 and PM-2 here. The particulate correction experiments use PM-1, which deposits, not the sum. The budgets 3M and 300k are nominal. At 3M, $\mathrm{NO}_x$ uses $3.00 \times 10^6$ particles, PM-1 $2.85 \times 10^6$ and PM-2 $3.00 \times 10^5$. At 300k, PM-1 uses $2.85 \times 10^5$. Labels are stored for the 0–3 m layer and in 3-D, on 5 m horizontal cells, domains 700 to 2,500 m wide and 42 layers up to 1,500 m.

The frozen split has 16,800 training, 3,600 validation and 3,600 test scenes. The deployment and the weighted-loss experiment use a balanced subset of 6,000 training scenes, 250 per combination. Scenes whose support above 9 m is empty (Appendix A) are then excluded by a rule fixed in advance, without redrawing the subset, which leaves 5,995 $\mathrm{NO}_x$ and 5,992 PM-1 scenes. The stored files record each run's particle counts, seed and code fingerprints.

## D.2 MONTE CARLO LABELS AND GENERATOR CHECKS

The GPU transport solver ports the AUSTAL 3.3 model from its source code (Ingenieurbüro Janicke, 2024). Repeated runs of one scene keep its geometry, emissions and weather and change only the MC seed. Within a run, the particles are dealt in turn to 100 subgroups, and ten consecutive subgroups form each of ten groups of equal size. The dose $d_{g,i}$ of group $g$ in cell $i$ accumulates the mass of its particles multiplied by the time they spend in the cell. From the doses of $n_g$ groups, the simulator reports

$$s_i = \sqrt{\max\left(0,\ \frac{1}{n_g - 1}\left(\frac{n_g \sum_g d_{g,i}^2}{\left(\sum_g d_{g,i}\right)^2} - 1\right)\right)}, \tag{17}$$

with $s_i = 0$ when the total dose is zero. This is the usual relative standard error of the mean of independent group estimates, written with the doses. The concentration $c_i$ is proportional to $\sum_g d_{g,i}$, and $s_i$ is its relative standard error, its standard deviation over its value. So $(s_i c_i)^2$ estimates the variance of cell $i$ alone. The correction reads the 3-D field, whose $s_i$ uses the ten groups, $n_g = 10$, even in the lowest layer. The stored 0–3 m layer also carries an $s_i$ from the 100 subgroups, which the label correction does not use. The simulator gives no covariance between cells, so the correlations of $\Sigma_N$ are modelled separately (Section 4.1). Appendix C.2 checks $s_i$ against the spread across runs.

Each group of a 3M run is itself a 300k estimate, nested in the 3M one and correlated with it. The learner test of Section 5.3 uses one group as the 300k label, so its two arms share particles. The deployment instead needs a label independent of its full-budget run, and simulates it as a separate run with another seed. Variances move between budgets by the $1/N$ law.

**Generator validation.** A verification suite, modelled on AUSTAL's own tests, checks the well-mixed condition (Thomson, 1987), under which evenly spread particles stay evenly spread, the growth of a plume's width with travel time (Taylor, 1922), the concentration profile of an elevated source (Berlyand, 1975), the rise of a hot plume above its source, dry deposition, settling, the downward motion of particles under gravity, and washout, their removal by rain. Separate runs compare the 0–3 m field with AUSTAL's on the same geometry, sources and weather. They use the share of cells within a factor of two (FAC2), the fractional bias and the normalized mean squared error. AUSTAL is the reference of the port, so these runs test its fidelity, not physical truth, and do not calibrate its MC error.