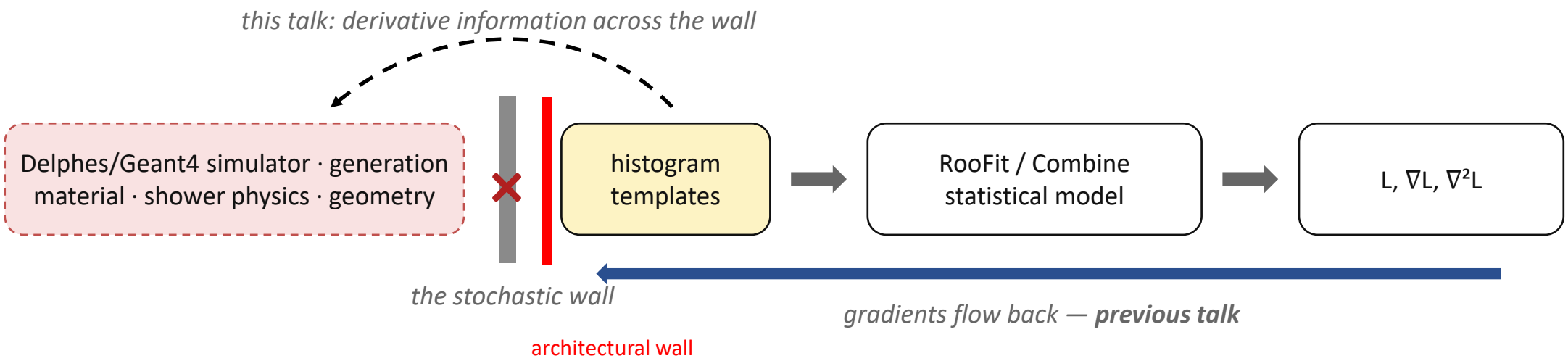
Trust, then scale: checkable gradients in a minimal end-to-end pipeline

Jonas Rembser*, Vassil Vassilev+
{ *CERN, +Princeton/compiler-research.org }

This work is partially supported by National Science Foundation under Grant OAC-2311471

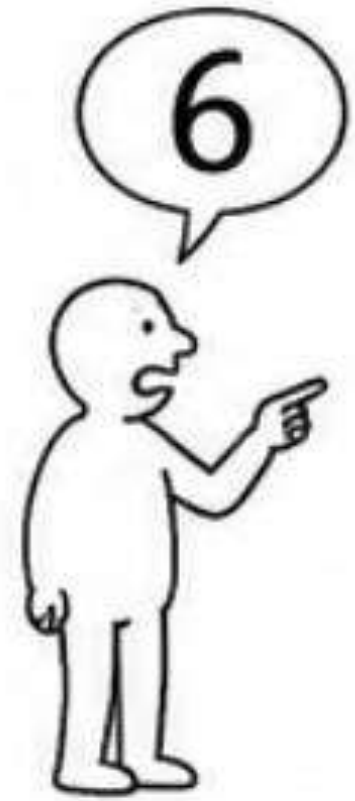
Some talks ago: the bottom half became real

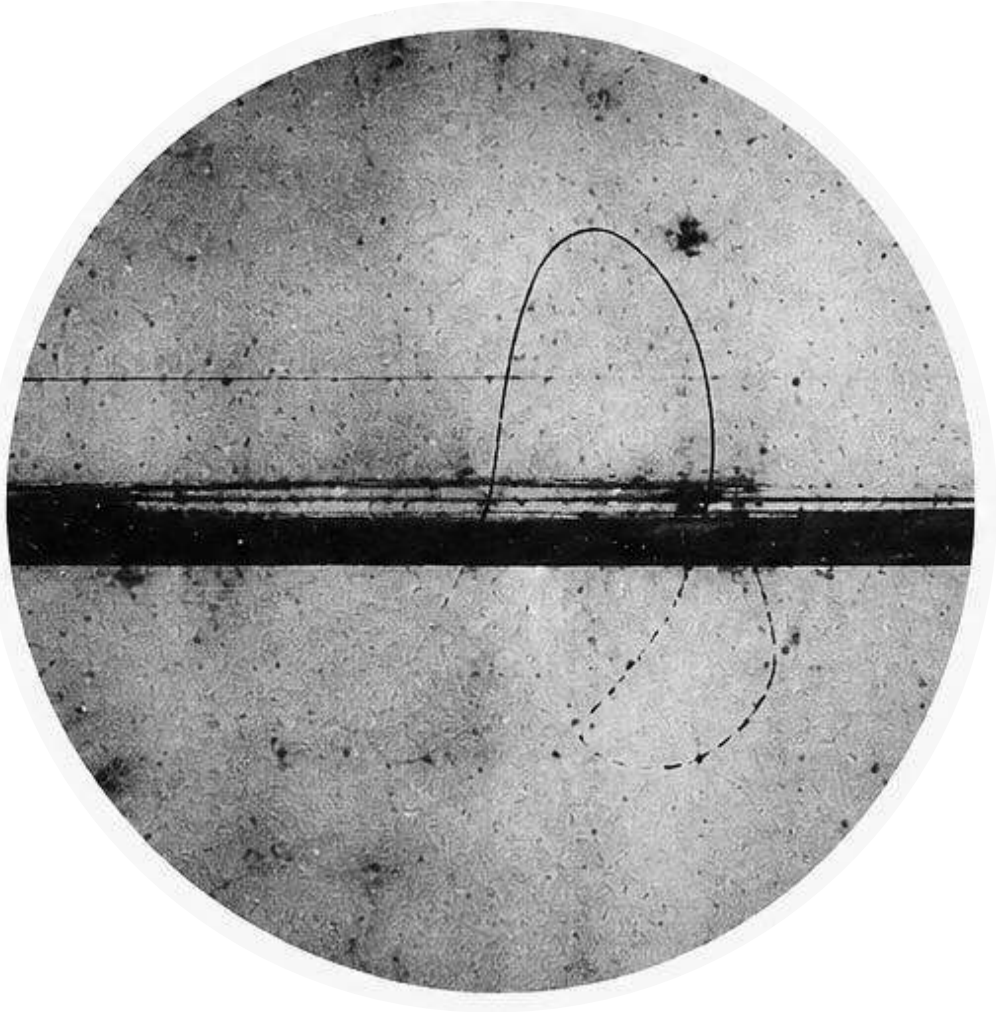


Yesterday's talk: AD in CMS Combine went from feasibility to capability

- The **analytic half of an end-to-end derivative chain** now exists in production tooling

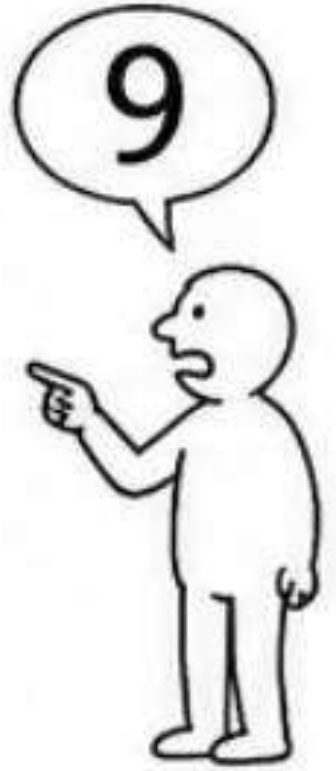
If you just walked in: previous talk = differentiable by construction; this talk = differentiable by measurement.





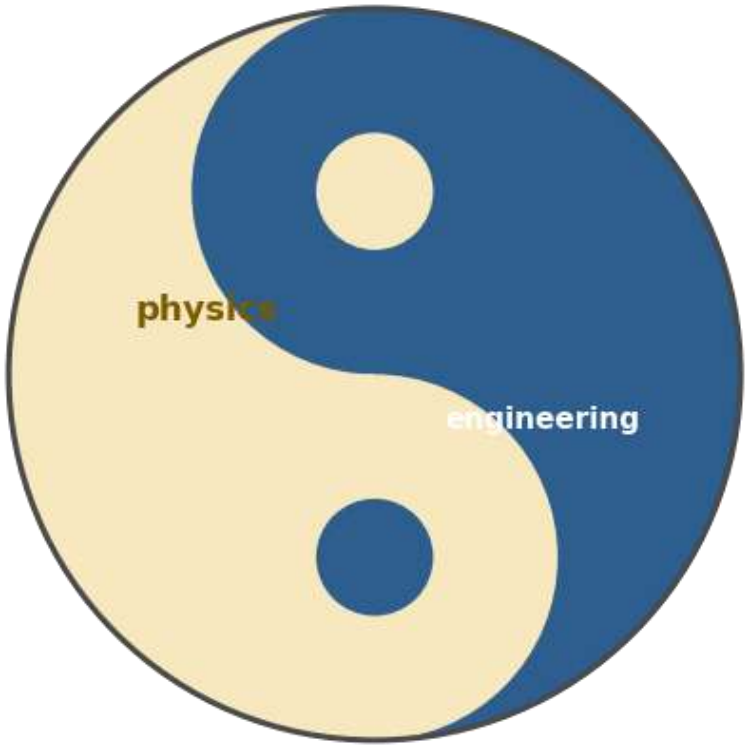
Carl Anderson, 1932

Positron (e^+) track in cloud chamber



Disagreements often start with language, not facts. Translate terms. Align models. Build understanding.

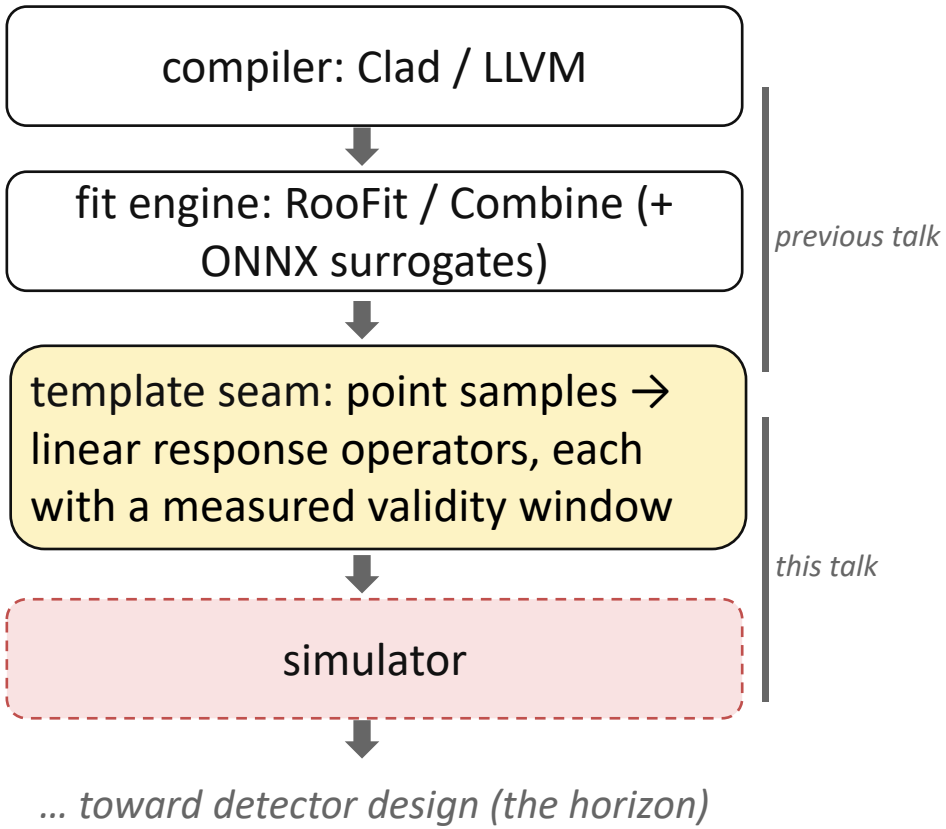
Two crowds; one goal: zen



the boundary between them is not a wall — it is the contract (T1–T6)

- Two audiences, one story: ○ marks the physics voice, ● the engineering voice; paired blocks say the same thing in both languages.
- **No claim without a check**: every number here reruns in ~14 s from one C++ file — github.com/vgvassilev/minlhc. Its six built-in checks, T1–T6, print pass/fail on every run; they are the falsifiers quoted throughout.
- The middle of the talk is a **formal case study**: research questions posed before the evidence, each naming its falsifier in advance.
- Inside the engineering half, a seed of physics: error bars, validity domains. Inside the physics half, a seed of engineering: a runnable pass/fail check.

A differentiable pipeline



Nobody planned this as one program. The segments compose anyway, because each ships **derivative information across a contract-guarded interface**. What this enables:

- **Systematics become measurements.** Buried nuisances become jointly estimable — keeping in check the systematics that dominate the HL-LHC era.
- **Simulation becomes reusable capital.** Extract operators once, serve many analyses, refresh when the window says so.
- **Design becomes optimization.** The same machinery can one day rank detector-design choices by physics reach.

None of this is claimed today. It is what is at stake which is exactly why the rest of this talk is about trust.

HEP is quietly rebuilding its software stack around derivative information as a first-class interface — and these two talks are two segments of one backbone.

The catch: at scale, “works” and “appears to work” are identical

Every promise on the previous slide dies the same death if the gradients are silently wrong. At full scale, nobody can tell:

○ For the physicist

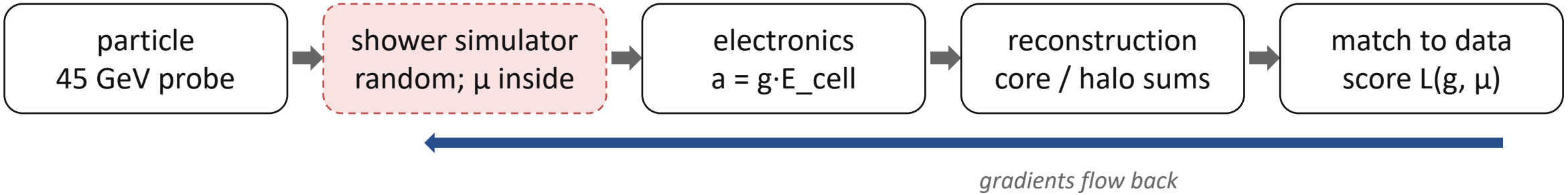
- ground truth unavailable
- brute-force baseline unaffordable
- failure modes hide in the stack
- huge upfront investment

● For the engineer

- no test oracle
- no reference implementation
- silent wrong answers
- months of work before the first signal

Prove the method where proof is possible; only then pay for scale. The rest of this talk is that case study.

A tiny, textbook pipeline (mostly) on one screen



- A particle splashes energy into a grid of cells; how *wide* the splash is, is set by μ (in real life: the Molière radius, fixed by the detector material). The splash is random — every event differs.
- The electronics multiply each cell by a gain g (real life: an ADC-per-GeV constant *we* must calibrate).
- We score a central 3×3 "core" and its "halo" ring: their **sum** constrains g, their **ratio** constrains μ — one scalar E_rec would leave an unfittable g–μ ridge.

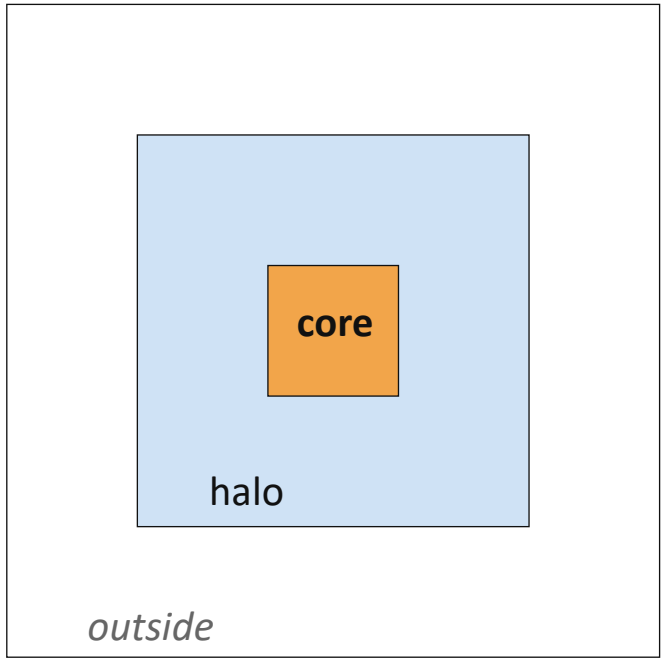
~1000 lines of C++, one file, no dependencies, runs in ~14 s. Two knobs: g is ours, μ is nature's. Truth planted at (1.00, 0.85); the wrong nominal (0.85, 1.00) is where the joint fit starts — and where the classical scan pins μ.

The buried knob: μ

- **What it is.** μ is the lateral width of the shower — the toy's stand-in for the Molière radius, set by the material the shower traverses. Nobody *chooses* μ: it is a property of the as-built detector that the simulation must *describe correctly*.
- **Why it is buried.** It manifests only through simulated showers — no closed form connects it to the likelihood. It lives inside the simulator's transport, like every material and shower-physics parameter at a real experiment.
- **Why it matters.** If the material description is wrong, the predicted shower *shape* is wrong — and the energy calibration silently absorbs the error: a wrong μ produces a wrong g. This entanglement is real — it is the material-budget systematic of every e/γ calibration — and it is the coupling this whole talk turns on.
- **Why it is not fitted today.** Without a gradient it can only be measured in dedicated side studies, frozen into the geometry, and wrapped in an uncertainty band — never co-estimated with the calibration it contaminates.

Our injection: truth μ = 0.85 vs. assumed nominal 1.00 — a simulation that over-estimates the lateral spread, i.e. a mis-described material budget.

Two observables, or the fit is impossible



- g multiplies *every* cell: E_core and E_halo move together.
- μ widens the splash: energy *migrates* from core to halo. The sum barely moves; the **ratio** moves.
- So: (core + halo) constrains g; (core / halo) constrains μ: two parameters, two observables. A single scalar E_rec could not constrain both.
  - *This is exactly the role shower-shape variables (R_9-like) play in a real calorimeter.*
  - *You cannot identify two parameters from one output; add an output that responds to only one of them.*

Research questions — posed before the evidence

	question	designated falsifier
RQ1	Can a gradient be obtained for a parameter inside a stochastic black-box simulator, without differentiating the simulator? • *Can we learn which way the answer moves when a hidden knob turns — without opening the box?*	T1 vs. brute force
RQ2	Is the extracted operator an artifact of one extraction, or stable? • *If we measure that response twice, independently, do we get the same answer?*	T2, disjoint streams
RQ3	Where does the frozen linearization stop holding — measured, not assumed? • *How far can the cheap shortcut be trusted — with a number on it?*	T3 window
RQ4	Is a single successful recovery sufficient evidence that the estimator can be trusted? • *One experiment got (nearly) the right answer — does that mean the method works?*	50 toys + scans
	Stated, not asked — by construction:	
Demo	the joint fit reaches a knob that fix-and-scan cannot even represent	T4 vs. T5 control
Property	correctness is checkable continuously, at negligible cost	the contract itself

The obstacle: a knob inside a black box

○ For the physicist

μ lives inside the Monte Carlo shower. There is no formula for ∂L/∂μ — the shower is a stochastic process, and each run gives different hits.

● For the engineer

The function calls the random-number generator internally. Blind Autodiff through it is either impossible (it is a black box) or meaningless (you would differentiate one noise realization).

We refuse both popular fixes:

- differentiating the simulator (O(10–100×) cost) or replacing it with a learned surrogate (a training domain that exits silently).
- Mature fields hit this wall decades ago and all made the same move: wing design solves *one adjoint equation for the drag* instead of differentiating the airflow; weather assimilation fits with a *local linearization of the forecast model, refreshed each cycle* — in production for thirty years; rendering differentiates *pixel expectations*, not photon paths.

Inference needs derivatives of a few observables — so we build only those.

Trick 1: same randomness, two runs

```
                                for (int s = 0; s < nSeeds; ++s) {
  std::mt19937 rng_p(seed + s), rng_m(seed + s);   // identical seeds!
  auto hp = simulate_shower(probe, mu0 + eps, rng_p, ...);
  auto hm = simulate_shower(probe, mu0 - eps, rng_m, ...);
  for (cell : cells)  sum[cell] += E(hp, cell) - E(hm, cell);
}
J[cell] = sum[cell] / (2 * eps * nSeeds);   // eps = 0.02, nSeeds = 1000
```

○ For the physicist

Common random numbers: correlated MC variations, the same idea behind correlated systematic-variation samples.

● For the engineer

An A/B test where both branches get identical traffic.

Deterministic replay makes an O(ε) effect measurable under O(1) noise. That is: pin the random seed, run the simulator at μ+ε and μ−ε with identical random draws, subtract. The noise cancels; what survives is how each cell's mean energy responds to the knob.

The noise is identical on both sides and cancels in the difference; what survives is the response of each cell's mean energy to the knob. A number per cell, J[cell] = ∂⟨E_cell⟩/∂μ, is J, the response Jacobian (239 non-zero cells in the toy).

Trick 2: extract once, freeze, splice into the chain rule

$$\frac{\partial L}{\partial \mu} = \sum_i \left(\frac{\partial L}{\partial A_i}\right) g\, J_i \qquad \frac{\partial L}{\partial g} = \sum_i \left(\frac{\partial L}{\partial A_i}\right) E_i$$

A_i = g·E_i — the digitized cell amplitudes the score reads; L — the match-to-data score (a toy Gaussian loss here; at scale, your analysis likelihood)

```
            double dL_dEc = -(E_core - target_core) / (sigma * sigma);  // analytic
for (auto& [cell, dEdmu] : J.dE_dmu) {      // sweep J's support
  double dL_dE = in_core(cell) ? dL_dEc : dL_dEh;
  grad_g  += dL_dE * E_true[cell];          // fully analytic path
  grad_mu += dL_dE * gain * dEdmu;          // frozen-operator path
}
// (event cells outside J's support still contribute to grad_g)
```

○ For the physicist

A local response matrix: adjoint physics, not differentiable simulation.

● For the engineer

A memoized linearization of an expensive service, spliced into backpropagation as one extra factor.

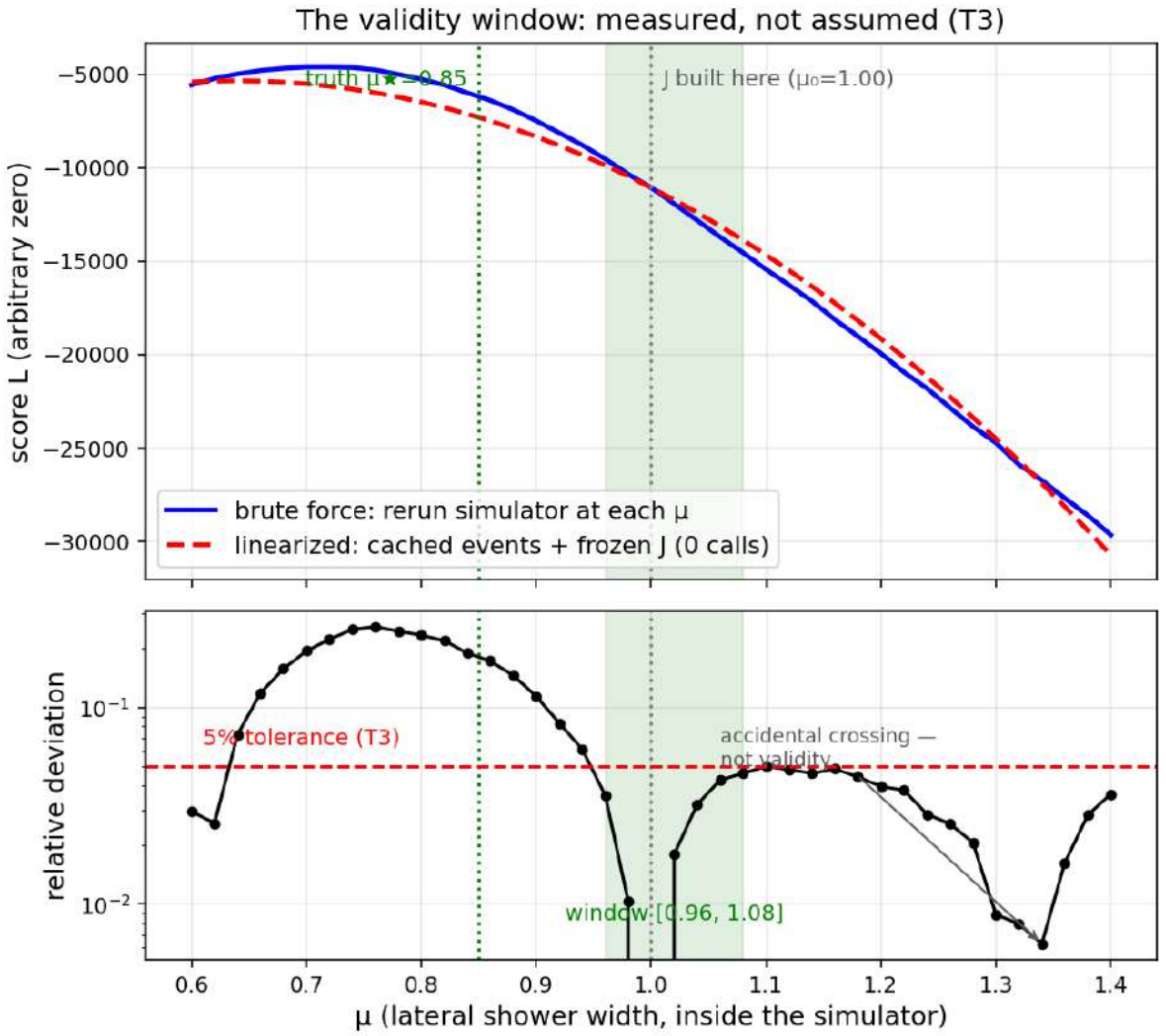
Everything downstream of the simulator is plain analytic code we differentiate by hand; the frozen J bridges the gap.

RQ1 & RQ2: the gradient exists, tracks brute force, and is stable

- **T1a (∂L/∂g, fully analytic path).** FD vs. the hand-derived adjoint: **9.9×10^{-13}** — machine precision. A coding check, not a discovery; at full scale this check is unavailable.
- **T1b (∂L/∂μ, through the frozen J).** Agreement **0.20** (tol. 0.25): the FD side reruns the stochastic oracle, the adjoint uses a 1000-seed estimate, with matched ε so both measure the same secant. Measured scaling (backup): the extraction component shrinks like 1/√N, saturating at a ~0.09 dataset floor.
- **T2 (stability).** A second J from a disjoint seed stream agrees on the downstream *prediction* to **0.36%** — low variance, **not correctness**; correctness is RQ3's job.

RQ1 answered: yes, with a quantified, shrinkable error; RQ2 answered: yes — deliberately not oversold.

RQ3 — the validity window is measurable, and narrow (T3)



Two ways to evaluate L(μ):

- **brute force**: rerun the simulator at every μ;
- **linearized**: cached events + frozen J — *zero* simulator calls.

Top: the curves hug at μ_0, then peel apart. Bottom: their gap — under the 5% tolerance only inside the shaded window **μ ∈ [0.96, 1.08]**.

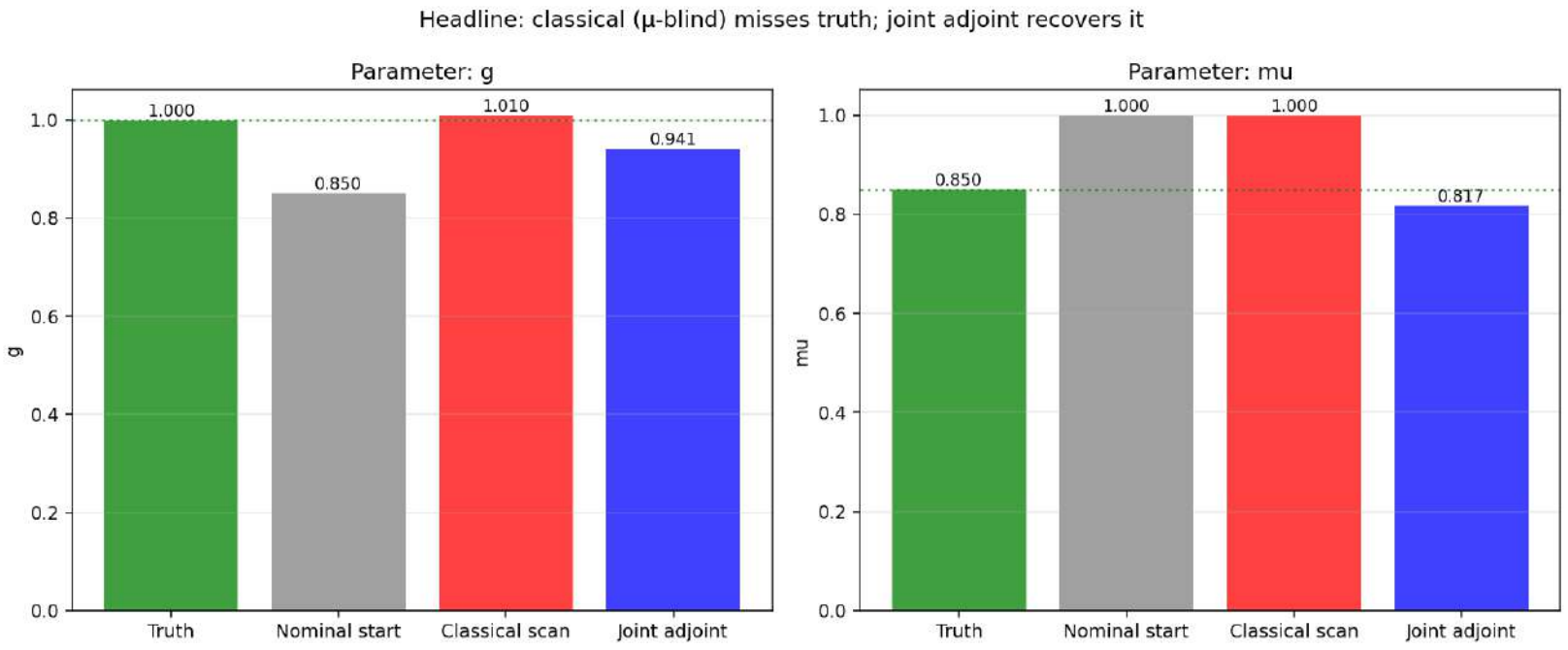
At the truth (μ = 0.85) the first J is already ~15% off — the refresh rule (every 50 steps) is not optional.

- *Cache invalidation with a measured TTL.*
- *The domain of validity of linear response.*

Still a parabola? Yes: L is quadratic in the energies. We linearize E(μ), not L(μ).

RQ3 answered: yes — the boundary of trust is a measured quantity, and it forces a refresh rule rather than a hope.

Demo: the joint fit reaches a knob the scan cannot (T4/T5)



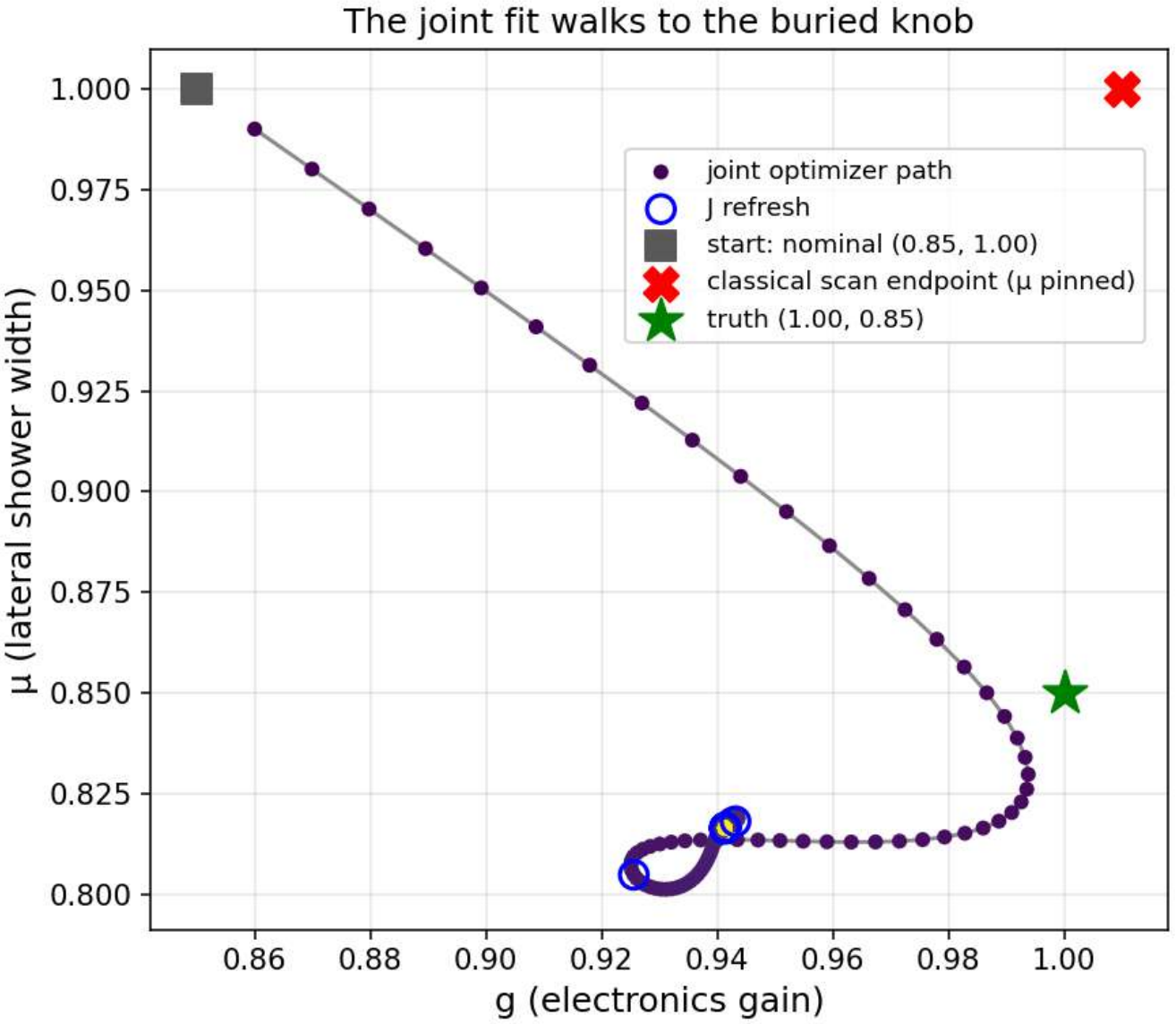
- **The joint fit turns both knobs at once**, guided by the cheap gradients (the frozen J; the simulator is never asked for derivatives). From the wrong start (0.85, 1.00) it walks to **g = 0.94, μ = 0.82** — close to the planted truth (1.00, 0.85). *(Adam, 800 steps, J refreshed every 50.)*

- **The old way, as the control (T5):** lock μ at the wrong guess and try every g. It finds a nearly perfect gain, g = 1.01 — and still fits far worse (score −8275 vs. −4492), because **no gain can compensate for a wrong shower width**. And it never even asks what μ is: that question is not in its vocabulary.

The baseline didn't return a worse answer to the question; it cannot ask the question. That, not the 1%, is the point.

Why isn't the recovery exact? Not statistics — two slides ahead, the ensemble traces the leftover offset to a flaw in the score itself (RQ4)

The demonstration in one picture



Joint fit of (g, μ) via the frozen J:

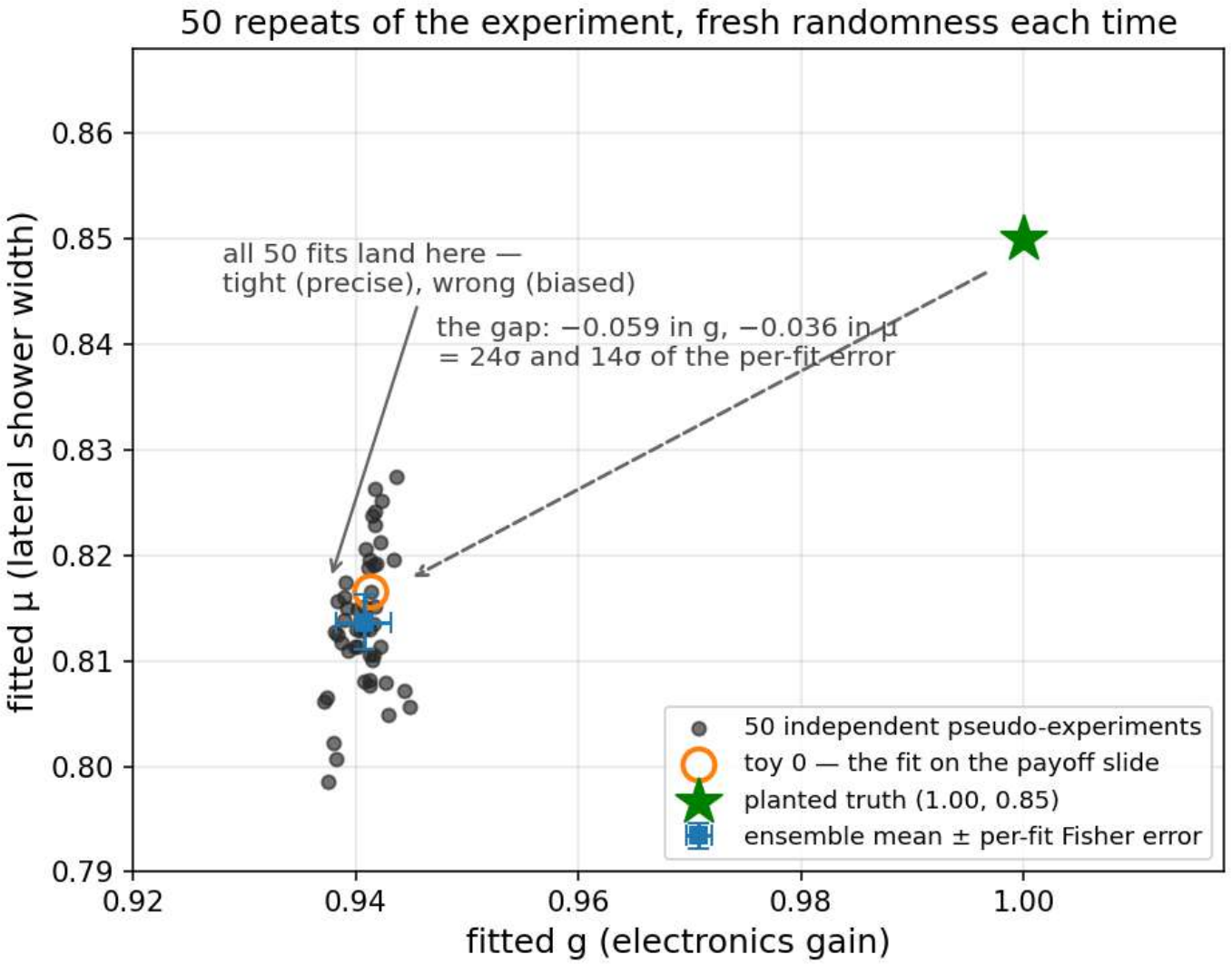
g = 0.94 (true 1.00);

μ = 0.82 (true 0.85)

- *Optimizing one config variable while the other is hard-coded to a wrong constant. The fit looks fine; the answer is not.*

The classical alternative: fix μ at its nominal value, scan g — cannot estimate μ *at all*: it is not in the search space.

RQ4 — so we ran it 50 more times



Fresh data and fresh model randomness in every repeat — the fit you just saw is **toy 0**, the orange ring.

	g	μ
ensemble mean	0.9407	0.8137
toy-to-toy scatter	0.0018	0.0065
per-fit Fisher error	0.0024	0.0026
bias vs. truth	−0.059	−0.036

All 50 land in the same wrong place. Tight cluster: the method is *precise*. Wrong place: it is *biased* — 24σ and 14σ of the per-fit error.

The single success was **precisely wrong** — and one run could never have told us.

RQ4 — diagnosis, fix, and the moral

- **Diagnosis.** The per-event score compares every event to the *average* target: $\Sigma_e(r_e - T)^2 = N[(\bar{r} - T)^2 + \mathrm{Var}\ r]$. The second term rewards being *steady*, not being *right* — and both knobs shrink the variance when lowered. The score's optimum is simply not at the truth.
- **A twist.** The optimizer's decaying step stops *short* of the score's own optimum (0.92, 0.76) — early stopping accidentally cancelled part of the bias. Two flaws, partially hiding each other.
- **The fix, annotated in the repo as T6.** Score the ensemble *mean* instead: the variance term vanishes by construction, and the optima return exactly to truth (1.00, 0.85). (scan figure: backup)
  - *The archer scored on summed arrow-to-bullseye distance: draw the bow less, group tighter, aim low — the score improves.*

RQ4 answered: no. One good-looking recovery proves little and the framework caught its own showcase. Silent wrongness, found by the estimator-level dual path.

Results Summary

RQ	check	plain meaning	result
RQ1	T1a	gradient math vs. brute force (g path)	9.9×10^{-13}
RQ1	T1b	gradient through frozen J vs. brute force	0.20, shrinks with seeds
RQ2	T2	two independent J's predict the same L	0.36%
RQ3	T3	where cheap tracks expensive	$\mu \in [0.96, 1.08]$
Demo	T4	recover an answer we planted	few-percent recovery
Demo	T5	the old way cannot ask	gap categorical
RQ4	toys	is one recovery enough?	no — bias $\gg$ Fisher
RQ4	T6	the fixed score is unbiased	optima exactly at truth



The cultural bridge in one artifact: physicists get stated domains of validity; engineers get CI-style pass/fail.

Risks

Internal / construct (does the evidence support the inference?)

- Closed world: answer planted, generated by the same model that fits it; in-sample targets; detector noise off.
- The mean-target score's variance penalty — found, measured, and fixed (RQ4/T6); matched-ε excludes the secant-vs-tangent artifact by construction; the shadow-cell bias was caught only by the dual-path check.
- The toy–CMSSW mapping is *structural*, not physical fidelity.

External (what may fail at scale — gate named in advance):

- Common-random-numbers survival in real Geant4 — gate: standalone-Geant4 stage.
- Window width at scale (refresh cost vs. structural saving) — gate: a measured window on one real observable.
- Observable sufficiency (real shower shapes) — gate: the physics pilot. Dimensionality: 239 cells vs. ~75k crystals.

Reliability: fixed seeds; independent runs bit-identical; environment, checksums, verbatim output in REPRODUCIBILITY.md.

The metric that substantiates the direction

Information return on simulation: Fisher information delivered per simulator call — *all costs loaded* (extraction, refresh, falsifier anchors), *at matched trust* (both workflows must pass the same falsifier suite before being compared).

tier	toy value (measured)	stays favorable iff
gradient machinery: operator calls vs. per-parameter FD	34k vs. 6.4M = 188×	the ratio grows with parameter count (structural)
break-even questions per window: extraction cost / per-question saving	N_break = 1	real usage asks ≥ N_break questions before each refresh
falsifier tax: validation calls / total calls	16.6% (printed by the run)	bounded as trust grows — classical revalidation is effectively 100%

The direction is substantiated while all three hold; a gate where one fails is a *measured* reason to stop.

- *Unit economics with the QA budget inside the unit — efficiency claims that exclude validation are the ones this talk exists to reject.*

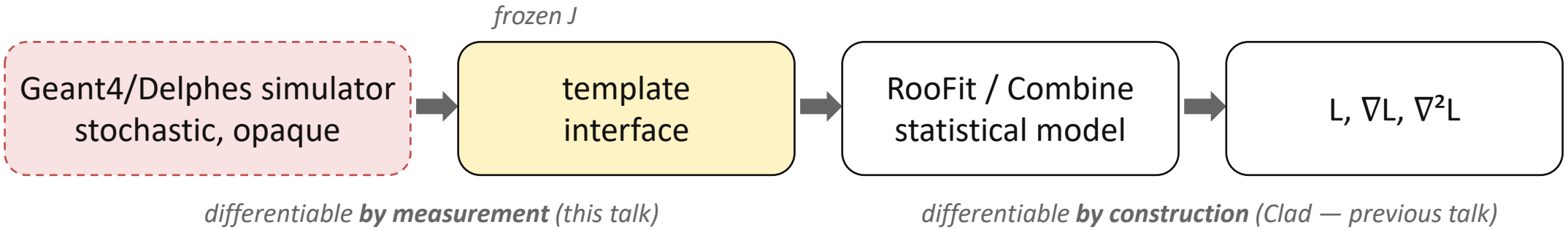
Doesn't the LHC already do this? Almost.

- **In-situ joint calibration fits.** ATLAS/CMS e/γ fits scale factors α_i and smearing terms c_i *jointly* in Z→ee, to ~0.05%. Genuinely joint, however, both are corrections applied *after* reconstruction. Nothing reaches inside the simulator.
- **Simulator-internal effects, handled sequentially.** Material budget is measured in dedicated studies, then *frozen* into the geometry; residual mismodeling becomes an uncertainty band. Bounded, never co-estimated — our T5 pattern, in production today.
- **Profile-likelihood analyses.** Hundreds of simulation-derived nuisances float jointly — via *template morphing:* bin-by-bin interpolation between histograms simulated at nominal and ±1σ anchors.

ATLAS e/γ calibration: arXiv:1407.5063, arXiv:1908.00005.

A morphed template histogram is a frozen linear response operator — fixed anchors, no measured validity window, no independence check, no refresh, no continuous simulator-internal knobs. We generalize something analyses already trust.

Zooming Out



- This is the opening slide's chain — but the stochastic wall now carries the frozen-J bridge, at exactly the spot where the × stood.
- The seam is not new infrastructure: it is where systematics templates *already* live.
- Composition is one chain-rule step: ∂L/∂θ_sim = (∂L/∂A)·J — and for information, **first-order J suffices**:

$$I(\theta_{sim}) = J^T I_A J$$

I_A = the likelihood's information at the level of the analytic inputs A — the AD Hessian yesterday's talk computes exactly

The thesis of both talks in one line: differentiate interfaces, not implementations.

Lessons Learnt

- **Silent wrongness is the failure mode.** Nothing here ever crashed; it was quietly wrong until a dual-path check said so. The cure is machinery, never vigilance.
- **Freeze, then price the freezing.** Every frozen object — the operator J, a template, a profiled parameter — must ship with its measured price (a window) and its refresh rule.
- **The contract is the transferable artifact.** What moves to CMSSW is not this code; it is T1–T6 as acceptance tests.
- **Objectives hide bias as easily as gradients do.** The 50-toy ensemble is the dual-path check at the *estimator* level: it re-attributed the recovery residual from "statistical floor" to a measured bias of the score function (the RQ4 result).

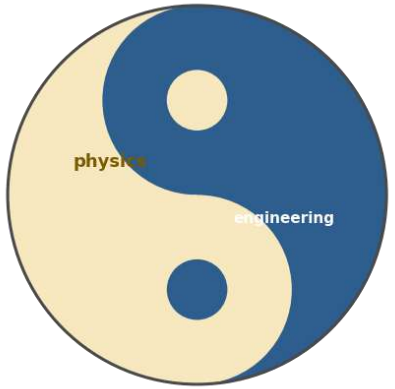
What the case earns: a small, checkable object to reason with before paying the cost of scale — and a de-risked, gated path for the application.

Takeaways

- **Physicist:** a nuisance buried inside the simulator can be fitted jointly with a calibration through a measured, frozen, refreshed response operator, where fix-and-scan cannot reach.
- **Engineer:** differentiate around the black box, not through it through deterministic replay + a memoized linearization + measured cache invalidation, guarded by fast pass/fail tests.
- **Teacher:** the whole argument fits on a screen, runs in 14 seconds, and every failure mode is a classroom exercise.

Which of these results survive contact with your detector, your reconstruction, your differentiable-programming stack — and which break first?

Thank you!

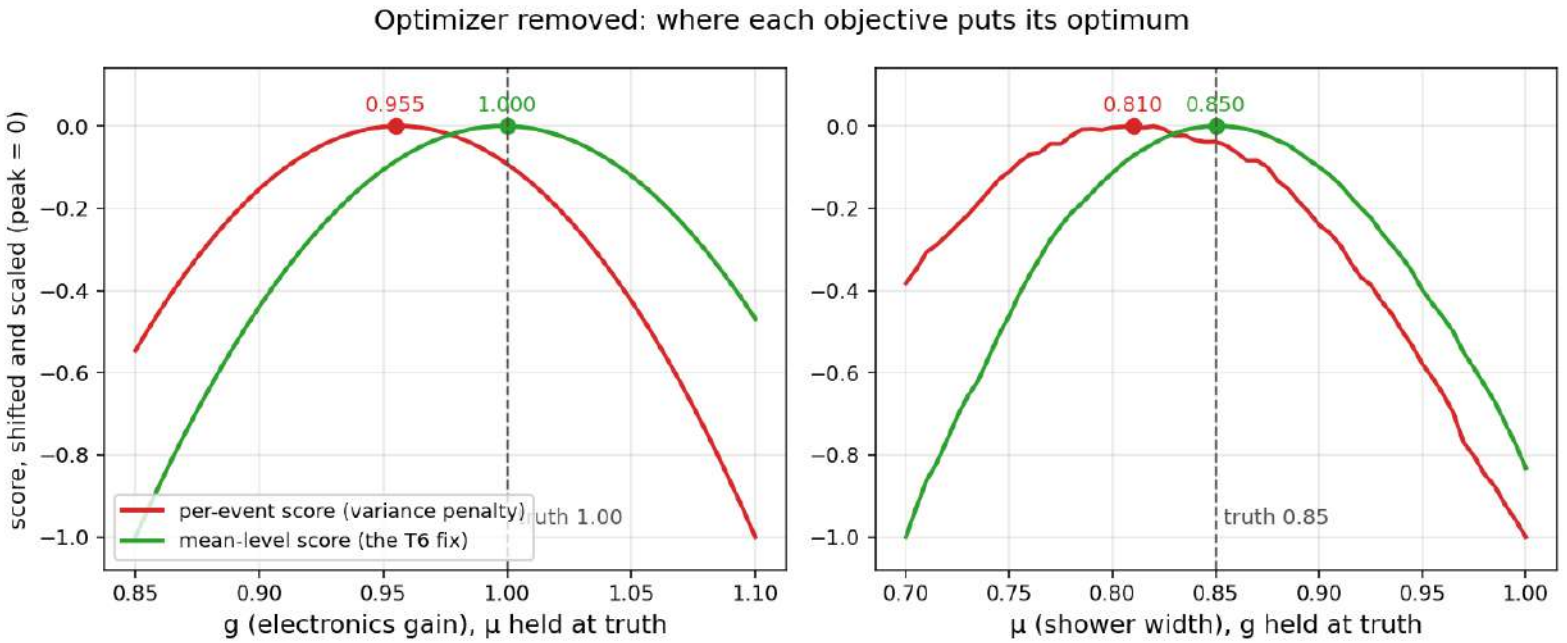


- MODE Collaboration, *Toward the end-to-end optimization of particle physics instruments with differentiable programming*, arXiv:2203.13818.
- P. de Castro, T. Dorigo, *INFERNO*, Comput. Phys. Commun. 244 (2019); arXiv:1806.04743.
- N. Simpson, L. Heinrich, *neos*, arXiv:2203.05570.
- G. C. Strong et al., *TomOpt*, arXiv:2309.14027.
- *Paraflow: fast calorimeter simulations parameterized in upstream material configurations*, Eur. Phys. J. C 85 (2025).
- T.-M. Li et al., *Differentiable Monte Carlo ray tracing through edge sampling*, SIGGRAPH Asia 2018.
- A. Jameson, *Aerodynamic design via control theory*, J. Sci. Comput. 3 (1988).
- F.-X. Le Dimet, O. Talagrand, Tellus A 38 (1986) — 4D-Var.
- ATLAS Collaboration, e/γ calibration: arXiv:1407.5063; arXiv:1908.00005.
- J. Rembser, V. Vassilev, *Automatic Differentiation in CMS Combine: From Integration to Physics-Grade Fits*, this workshop.
- Code and reproducibility record: github.com/vgvassilev/minlhc (Apache-2.0).

FAQ: glossary — HEP ↔ software

HEP term ○	**software term ●**
adjoint / backward pass	backpropagation
likelihood L	loss function (with error bars)
nuisance parameter	config value you must estimate, not set
calibration constant	config value you tune
Monte Carlo simulation	randomized black-box service
common random numbers	deterministic replay / seed pinning
response Jacobian J	memoized local linearization
validity window	cache-invalidation condition (measured TTL)
injection–recovery study	round-trip test against a planted answer
fix-the-nuisance-and-scan	grid search with a knob hard-coded
systematic uncertainty	known-unknown bias term
conditions database	config store with versioned constants
template (histogram)	precomputed table of expected bin counts
template morphing	interpolation between a few cached templates

FAQ: the fixed score, shown



Both objectives scanned on shared events along the truth axes (concept/bias_check.cpp).

The per-event score peaks off truth (g = 0.955, μ = 0.810). That is the variance penalty at work.

The mean-level score peaks exactly at truth (1.00, 0.85). The T6 fix, shown rather than printed.

FAQ: the method

- **“Isn't the T5 result true by construction?”** — Yes, deliberately: it demonstrates the workflow's *structural* blindness (μ is not in its parameter space). That is why it is labeled D, not RQ; the empirical content lives in RQ1–RQ4.
- **“Your recovery is biased — is the method broken?”** — No: the bias sits in the toy's *score*, not the operator machinery: gradients tracked brute force throughout (T1). The framework found, quantified, and explained it. The fix is now **T6 in the demo**: mean-level score optima return exactly to truth (1.00, 0.85) while the per-event score's sit at (0.96, 0.81) on the same events.
- **“What are the fit uncertainties?”** — Per-fit Fisher from J^T I_A J: ±0.0024 (g), ±0.0026 (μ). Ensemble scatter/Fisher: 0.74 (g), 2.5 (μ) — the μ under-coverage is stated, not hidden.
- **“Will matched seeds survive real Geant4?”** — Open; the named standalone G4 gate. To test: event-level pairing, control variates on the nominal, larger ensembles. Measuring the decorrelation *is* the first deliverable.

FAQ: scale and cost

- **“How does this scale to 75k cells × many parameters?”** — The operator is per *analysis observable*, not per cell: J of the handful of quantities entering the likelihood. Extraction cost scales with the parameters you probe, not with detector channels.
- **“What about noise and pileup?”** — Off by design here. Survival is not claimed — it is a listed threat with a gate, and the first classroom exercise.
- **“Does this reduce the need for simulation?”** — It reduces simulation *per question*, not the need: variation samples shrink (~10–100× in that category; PB of stored variants → MB of operators), the nominal anchor stays, and cheaper exploration raises total demand. The right metric: inferences per simulated event.
- **“So less money on CPU?”** — Not directly: pledge-funded computing runs full, so efficiency is reabsorbed, not refunded. The money case is the HL-LHC need-vs-flat-cash gap, variation-sample *storage*, and analyst time — while permanently paying a deliberate *falsifier tax* on brute-force anchors.

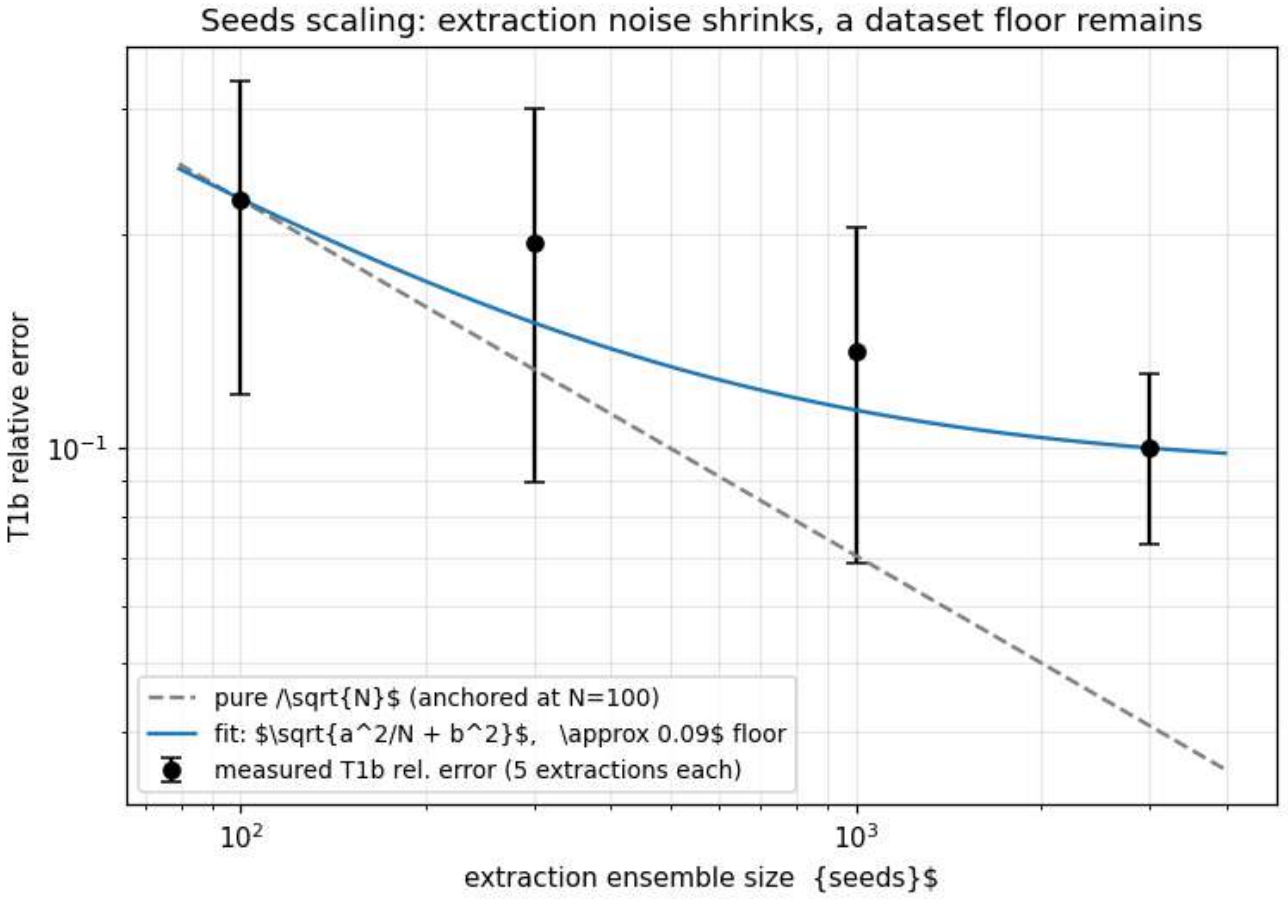
FAQ: Context

- **“Doesn't the LHC already do this?”** — Almost (main-deck taxonomy slide): in-situ (α, c) fits are downstream-only; material is handled sequentially; template morphing is our degenerate case — fixed anchors, no measured window, no independence check, no refresh.
- **“Why not a differentiable simulator, or a surrogate?”** — O(10–100×) cost, or a training domain that exits silently (backup: options considered). Inference needs only how a few observables respond to small parameter changes.
- **“Is μ the signal strength?”** — No: a shower width. Nothing in this talk profiles a POI.

FAQ: ensemble fine print

- Harness: concept/toy_ensemble.cpp — includes the tracked demo unmodified; toy 0 reproduces the talk's exact realization. Per toy: fresh data seed, fresh model-MC seed, the same optimizer settings as the talk.
- Per-fit errors from the 2×2 Fisher composition $I = J^T I_A J$ at the fit point — the deck's own identity, exercised on the toy.
- Fisher describes the g scatter (ratio 0.74) and under-covers μ (ratio 2.5; optimization + refresh noise contribute beyond statistics).
- Objective scans (concept/bias_check.cpp), optimizer out of the loop: g-scan at true μ peaks at 0.955; μ-scan at true g peaks at 0.810; joint grid optimum (0.915, 0.755), L = –4202 — the bias lives in the *score*, not the optimizer.
- Now banked as **T6** in the demo: scoring the ensemble *mean* deletes the variance-penalty term; its optima return exactly to truth (deviation 10^{-16}).

FAQ: seeds scaling of T1b



N_seeds	T1b rel. error	×√N
100	0.22 ± 0.10	2.2
300	0.20 ± 0.11	3.4
1000	0.14 ± 0.07	4.3
3000	0.10 ± 0.03	5.5

$$err(N) \approx \sqrt{\frac{a^2}{N} + b^2}$$

Five independent extractions per point. The pre-registered claim (“shrinks like 1/√N”) is **refined by measurement**: a ≈ 2.0, floor b ≈ 0.09 — the mismatch between an ensemble-mean operator and one finite dataset's own response, which no amount of extraction statistics removes.

FAQ: Small but stochastic

```
                std::vector<CaloHit> simulate_shower(const Particle& p, double mu,
                                     std::mt19937& rng, ...) {
  for (int l = 0; l < nLayers; ++l) {
    // longitudinal falloff (radiation-length analog), +-10% per layer
    double e_layer = p.E * std::exp(-0.3 * l) * fluct(rng);
    for (int h = 0; h < hitsPerLayer; ++h) {
      double u1 = unit(rng), u2 = unit(rng);   // standard normals
      hits.push_back({l, mu *u1, mu * u2, e_layer / hitsPerLayer});
    }
  }
}
```

- μ *multiplies standard normals*: pin the seed and the draws are identical — only the μ-dependence remains when you difference two runs.
- The randomness is real: hits land in different cells every event, and cells gain/lose energy discontinuously as μ moves them across cell boundaries. Differentiability is *not* smuggled in.

FAQ: the experiment we run on the testbed

- **Truth (planted):** g = 1.00, μ = 0.85 — a slightly narrower shower than assumed, correct gain.
- **Nominal (the assumed defaults):** g = 0.85, μ = 1.00 — wrong in *both* directions. The joint fit starts here; the classical scan pins μ here (its g range is unrestricted).
- **Data:** N = 2000 events at truth; targets are the mean core and halo energies. Detector noise off. σ = 10 in the score.
- **Task:** starting from nominal, recover both knobs — one of which no participant can differentiate directly.

● *An integration test with a planted answer.* • *LHC-jargon warning: μ here is a shower width, not a signal strength.*

FAQ: what the toy processes stand for

- **Longitudinal profile.** An e/γ cascades via bremsstrahlung and pair production; deposition falls on the radiation-length scale X_0. Toy: per-layer $E \propto \exp(-0.3\,\ell)$, ±10% layer factor for sampling fluctuations.
- **Lateral spread.** Multiple Coulomb scattering; characterized by the Molière radius (~90% containment). Toy: Gaussian transverse profile of width μ. Since R_M depends on the upstream material description, μ is a *genuine* buried nuisance.
- **Gain g.** Scintillation/ionization → ADC conversion, held in conditions data. A wrong g rescales all cell energies *coherently*.
- **Matched seeds.** Common random numbers: the shower becomes a deterministic function of (μ, seed), so differencing isolates $\partial\langle E_cell\rangle/\partial\mu$ instead of drowning an $O(\varepsilon)$ signal in O(1) noise.

FAQ: trick fine print and shadow cells

Choosing ε and the ensemble:

- ε = 0.02: large enough that hits cross cell boundaries between the runs (the response *is* that migration), small enough to stay linear.
- 1000 seed pairs; the probe ensemble is deliberately *independent* of the analysis dataset. Result: 239 cells with non-zero response; core cells negative, neighbors positive — the operator contains the physics picture.

Shadow cells:

- J can be non-zero at cells with no energy in the current event — they *would* light up if μ grew. The adjoint sweeps the **support of J**, not of the event; forgetting them is a silent bias, caught only because T1b compares against brute force.

● *When you diff two sparse maps, iterate over the union of keys, not the intersection. Fixture independence: never extract your approximation from the data you will test it on.*

FAQ: the price sheet

Paid. One operator extraction = 2000 oracle calls (1000 seed pairs × 2). The full fit performs 17 extractions (~34k probe showers) plus one forward pass over the 2000 events per iteration. Total wall clock: ~14 s — and the run prints its own oracle-call ledger (1.96M calls; falsifier share 16.6%).

Saved (structurally).

- The gradient of *every* parameter comes from one forward pass + the frozen J — no per-parameter re-simulation. A grid scan grows exponentially with parameters; per-parameter FD grows linearly.
- Inside the validity window, L(μ) is evaluable from cached events with *zero* simulator calls.

Not claimed. Any measured wall-clock speedup on a real workload. The efficiency argument is structural; benchmarking on something real is future work, stated as such.

FAQ: options considered, and rejected

Fix A: differentiate the simulator

Technically possible, but stochastic branching, deep control flow, and memory for the backward pass cost O(10–100×) — impractical for production workflows.

Fix B: replace it with a surrogate

Global surrogates struggle outside their training domain and are hard to validate to the standard detector experts require.

Both fixes make the *simulator* differentiable. But inference does not need that — it needs gradients of *physics objectives*, and the simulator enters those only through how a handful of observables respond to small parameter changes. Precedents for the third path: adjoint CFD, variational data assimilation, differentiable rendering.

FAQ: every toy stage has a real counterpart

toy stage	**CMSSW counterpart**	**deliberately dropped**
45 GeV probe	generator / particle gun	spectrum, event mix
shower simulator, μ	Geant4 (SimG4Core); μ ↔ Molière radius / material budget	full shower physics
a = g·E_cell	EcalDigiProducer + conditions DB	pulse shape, noise, pileup
core / halo sums	PF clustering; ratio ≈ shower shape (R_9-like)	dynamic clustering, tracking
match-to-data score	in-situ calibration (Z→ee), RooFit/Combine	full covariance, systematics

Transfers: *where the parameters live; the scale-vs-shape entanglement; seed-matched extraction.* ***Does not transfer:*** *dimensionality, and the affordable brute-force baseline — which is exactly why it must be established here.*

FAQ: what this approach adds

out there today	what it provides	what ours adds
template morphing (every profile-likelihood fit)	joint floating of sim-derived nuisances, 2–3 anchor points	continuous response; measured validity window; independence-checked extraction; refresh policy
sequential calibration (measure, freeze, band)	best-effort nuisance value + systematic band	the nuisance becomes jointly estimable; correlation with the calibration kept
in-situ (α, c) joint fits	precise downstream scale/resolution	reach upstream, into simulator-internal parameters
differentiable simulators	exact gradients everywhere	no O(10–100×) cost; trusted simulator runs unmodified
learned surrogates	fast parameterized forward model	no training domain to exit silently; validity measured per extraction

*The meta-benefit: **every number ships with the diagnostic that would falsify it.** Only brute force can say the same — and brute force is the one that does not scale.*

FAQ: the analytic half is production-grade

"Automatic Differentiation in CMS Combine: From Integration to Physics-Grade Fits" — yesterdays talk — reports Clad-based AD in RooFit/Combine: gradients, and newly, **Hessians**. Hessian of the CMS Higgs discovery likelihood (102 channels, 672 parameters):

method	**time**
numerical, default strategy	25 s
numerical, strategy 3 (robust)	1 min 23 s
Clad (exact)	3 min 3 s (+49 s one-time JIT)

- Read honestly: a *first implementation*, slower than numerical today; the fair comparison is strategy 3 (~2×, while exact — no step-size tuning, no failed error matrices). O(p) Hessian-vector products vs. O(p^2) evaluations shifts the crossover as models grow; JIT amortizes over toys, scans, rankings.
- What it hands *this* talk: T1a at scale is generated and validated by the compiler — and the I_A that the Fisher composition J^T I_A J needs (exact within its stated scope).

FAQ: same failure mode

At production scale (Combine AD)

A snapshot-style codegen prototype silently *zeroed* nuisance gradients hiding in mutable caches — “wrong answers, not crashes.” Caught by **dual-mode CI**: every reference test runs numeric and AD backends, auto-compared; “divergence fails the build, not the physicist.”

At toy scale (this testbed)

Forgetting shadow cells silently biases ∂L/∂μ; a stale J silently misleads outside its window; a mean-target score silently biases the estimator. All caught only by **dual-path checks** — T1b, T3, and the toy ensemble.

The defining failure mode of differentiable methods is **silent wrongness**. Two projects, at opposite ends of the scale axis, independently converged on the same instrument: continuous dual-path comparison built into the machinery, never left to vigilance.

FAQ: contracts are a coordination technology

- The Combine AD effort is a **bottom-up coalition** — compiler researchers, ROOT, CMS maintainers — with commits across all three repositories. “No single participant could have justified this alone; none paid more than a marginal cost.” It survived funding cycles and personnel churn because the shared object is a contract-guarded interface, not a monolith.
- The same contracts **admit new kinds of contributors safely**: GSoC students shipping code into a flagship CMS tool; AI-assisted scaffolding judged “acceptable *because dual-mode CI validates the output*.”
- “AD became a forcing function for software quality” — the refactors it demanded leave the tool better even where AD is off.

The technical architecture and the social architecture are the same architecture.

FAQ: Teaching

- The coupling between calibration and physics inference is hard to *see* in a production stack and easy to see here: change g, watch the likelihood move; change μ, watch core and halo trade energy. The whole causal chain fits on a screen.
- Every toy function is a signpost to a real subsystem — what you learn transfers as vocabulary, not just intuition.
- Exercises, each breaking a specific diagnostic: turn noise on; start outside the window and watch a stale J mislead; cut the extraction ensemble and watch T1b degrade; remove the halo observable and watch identifiability vanish.

- *It is a kata: small, deterministic, fast tests, and every mutation of the code maps to a visible test failure.*

FAQ: reproduce everything in under a minute

```
                    clang++ -std=c++17 -O3 concept/opaque_oracle_demo.cpp \
        -o opaque_oracle_demo
./opaque_oracle_demo      # ~14 s; prints T1-T6, the oracle-call
                          # ledger, pass/fail; exit code = failures
python3 concept/plot_oracle_demo.py   # figures in this deck
```

Environment, source checksums, verbatim recorded output, and a determinism check live in **REPRODUCIBILITY.md**.

Code: **github.com/vgvassilev/minlhc** (Apache-2.0).

FAQ: the testbed in numbers

- Geometry: 5 layers × 10×10 cells; core 3×3, halo 5×5 annulus per layer; 3 hits per layer.
- Truth $g = 1.00$, $\mu = 0.85$; nominal start $g = 0.85$, $\mu = 1.00$; $N = 2000$ events; noise off; $\sigma = 10$.
- J: 1000 matched-seed runs, $\varepsilon = 0.02$, 239 non-zero cells; refreshed every 50 optimizer steps (17 extractions total).
- Classical control: g-scan [0.70, 1.40] step 0.01 at $\mu = 1.00$; best $g = 1.01$, score −8275 vs. joint −4492.
- Core/halo ratio depends on μ but not g — the asymmetry that makes the pair identifiable at all.