Accelerating Uncertainty Quantification with Multifidelity Statistical Estimators

From Theory to Practice with PyApprox

60 minutes

John D. Jakeman
Principal Member of Technical Staff · Sandia National Laboratories

SAND2026-XXXXC · Exceptional service in the national interest

Accelerating Design and Certification

Make the design and certification of multiphysics, multiscale engineered systems faster and cheaper.

Experiment and simulation

Experiment grounds the models in reality; simulation reaches where tests can’t. Typically, neither alone is enough.

Uncertainty is unavoidable

Future loads, material scatter, the operating environment — all must themselves be estimated from data and simulation.

Accuracy and reliability are expensive

They are statistical claims under uncertainty — a failure probability, a defensible margin — so certifying them needs the full statistics, requiring thousands or more expensive simulations.

Multi-Fidelity Statistical Estimation

up to

1000×

fewer high-fidelity evaluations — for the same accuracy, without bias

Experimental data

Trusted high-fidelity simulation

AI surrogates

fused into a single unbiased statistical estimate

Roadmap

6 min	**Framing & the wing-spar benchmark**	Slides
8 min	**Monte Carlo: why variance is the hard part**	Slides
14 min	**Two models: control variates → ACV**	**Slides + Hands-on**
14 min	**Many models: GroupACV, the unifier**	**Slides + Hands-on**
10 min	**Extensions & the estimation cookbook**	Slides
8 min	**Results, pitfalls & wrap-up**	Slides

Five Ways Models Differ in Fidelity

Every member is an imperfect, correlated estimate of the same quantity. One member is the trusted high-fidelity reference the estimate is anchored to; the rest differ from it along five dimensions — and those five do not form a single ranked axis.

Geometry
What and how much is modeled — extent or shape.

Numerical resolution
How finely it is discretized — mesh or time step.

Physics
Which governing equations are solved.

Data-fit surrogate
The solver replaced by a model trained on data.

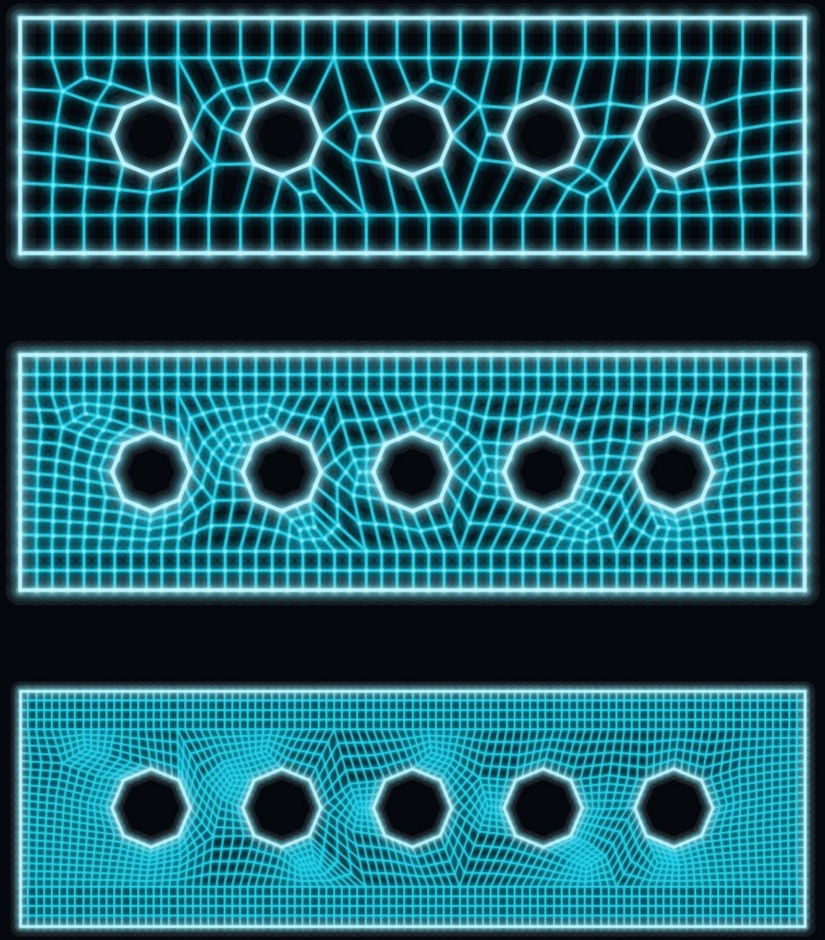
Source
Measured or computed — experiment or simulation.

Three reduce a simulation; one replaces it; one is where the evaluation comes from. The estimator ranks none in advance — only by cost and correlation.

Fidelity by numerical resolution

coarse

low cost



coarse

151 elem

medium

604 elem

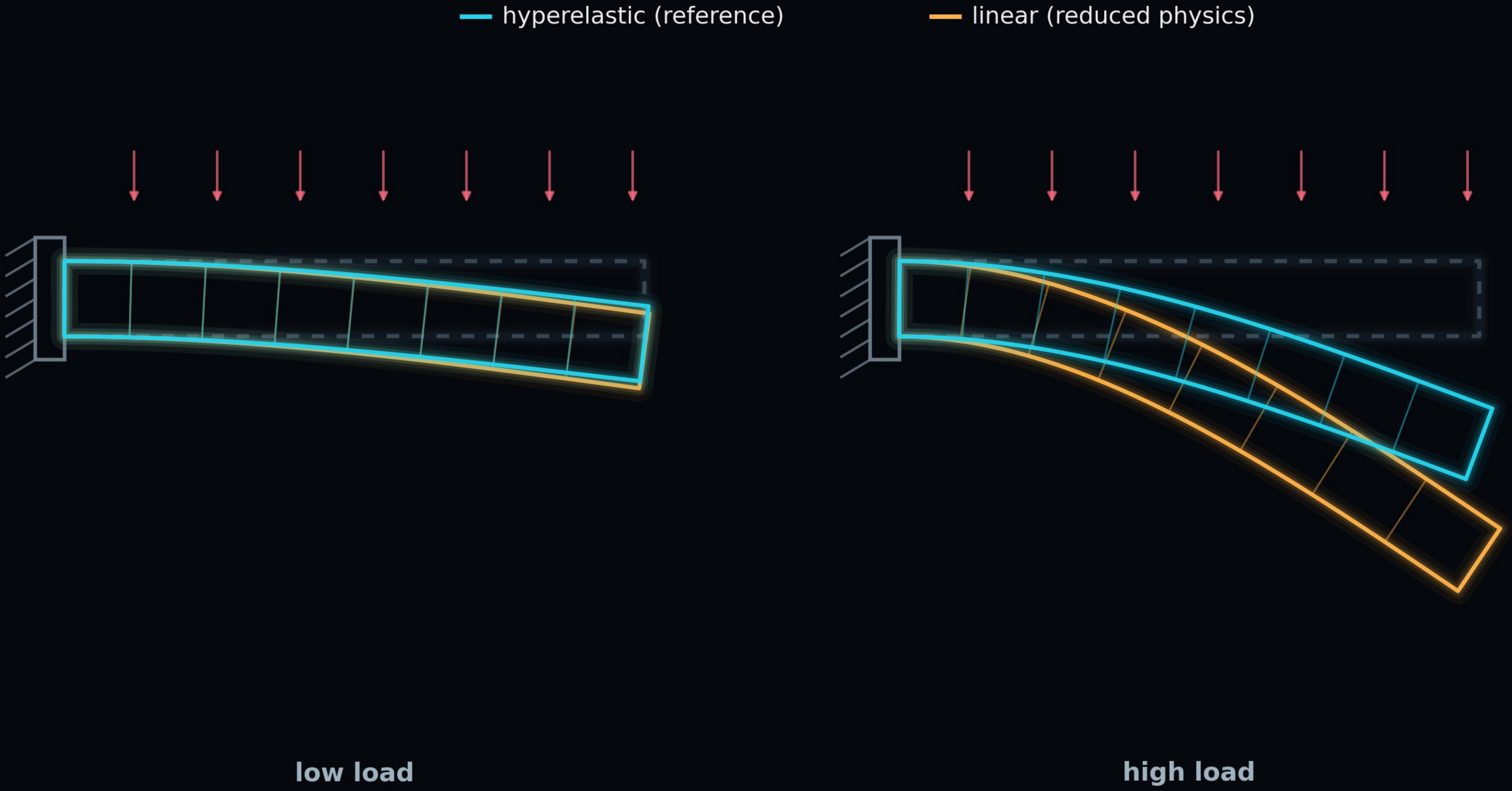
fine

2416 elem

fine

high cost

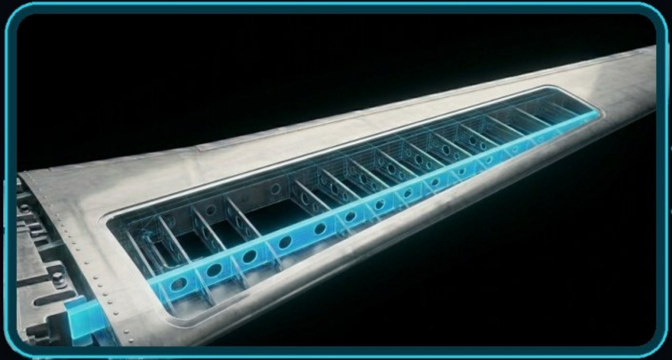
Fidelity by physics



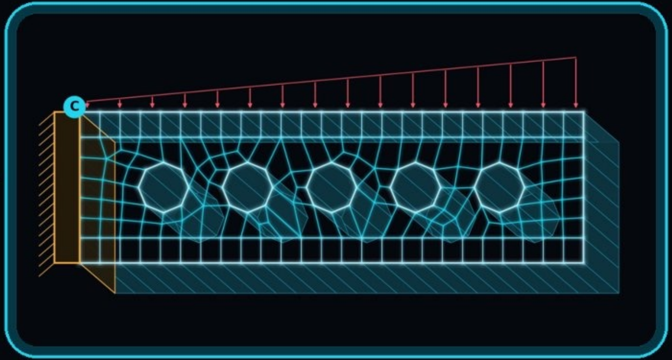
Fidelity by geometry

EXTENT

AIRCRAFT



WING BOX



BEAM

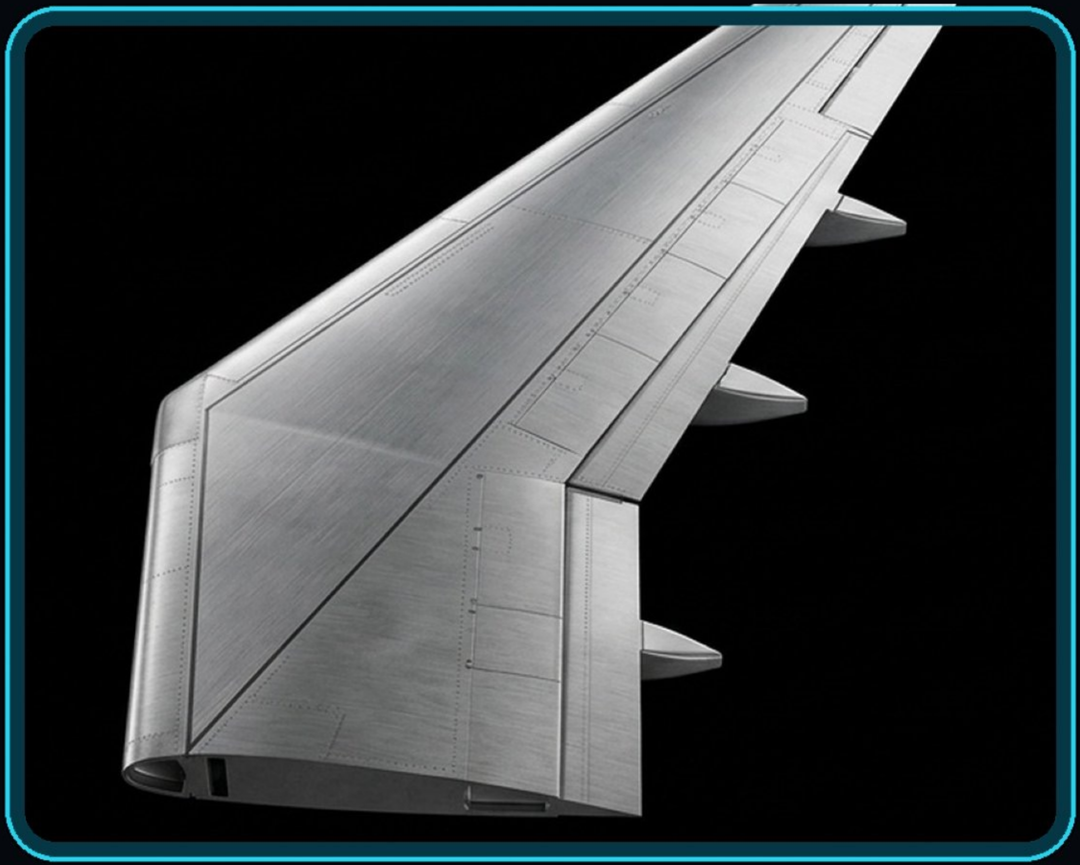
SHAPE

BASELINE

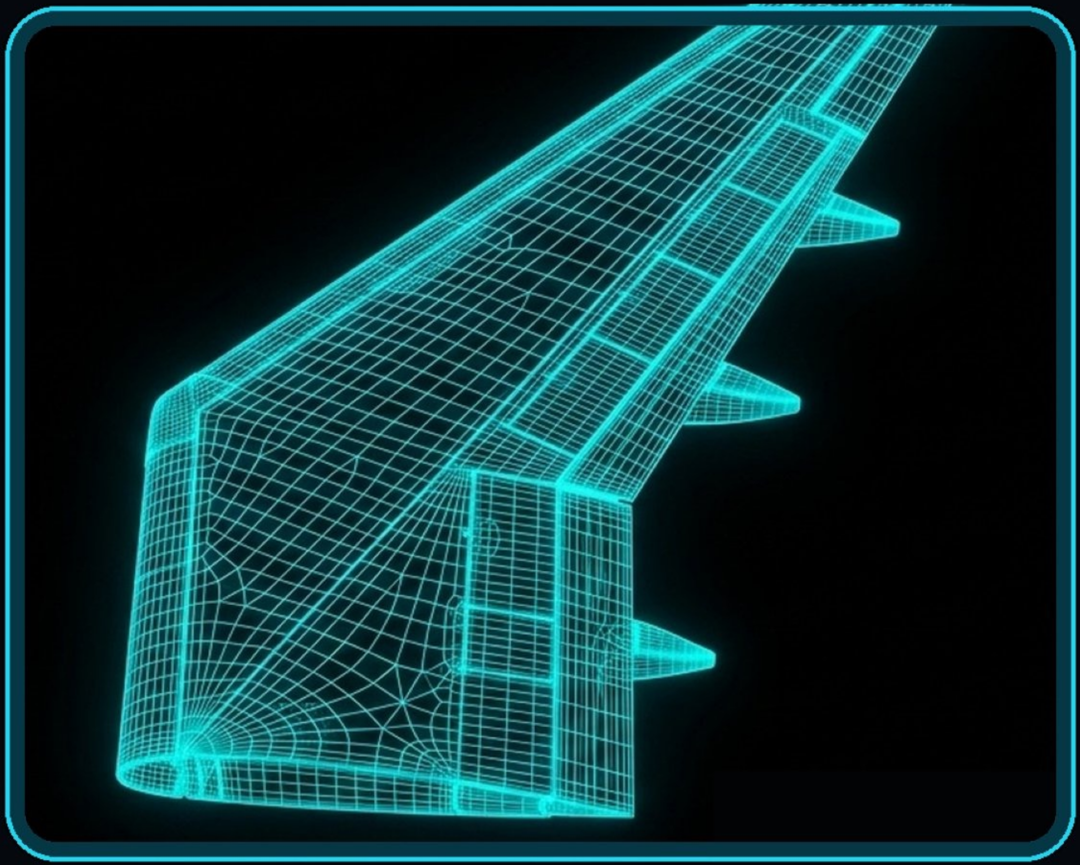
≈

RELATED / PREVIOUS DESIGN

Experiment and simulation



↔



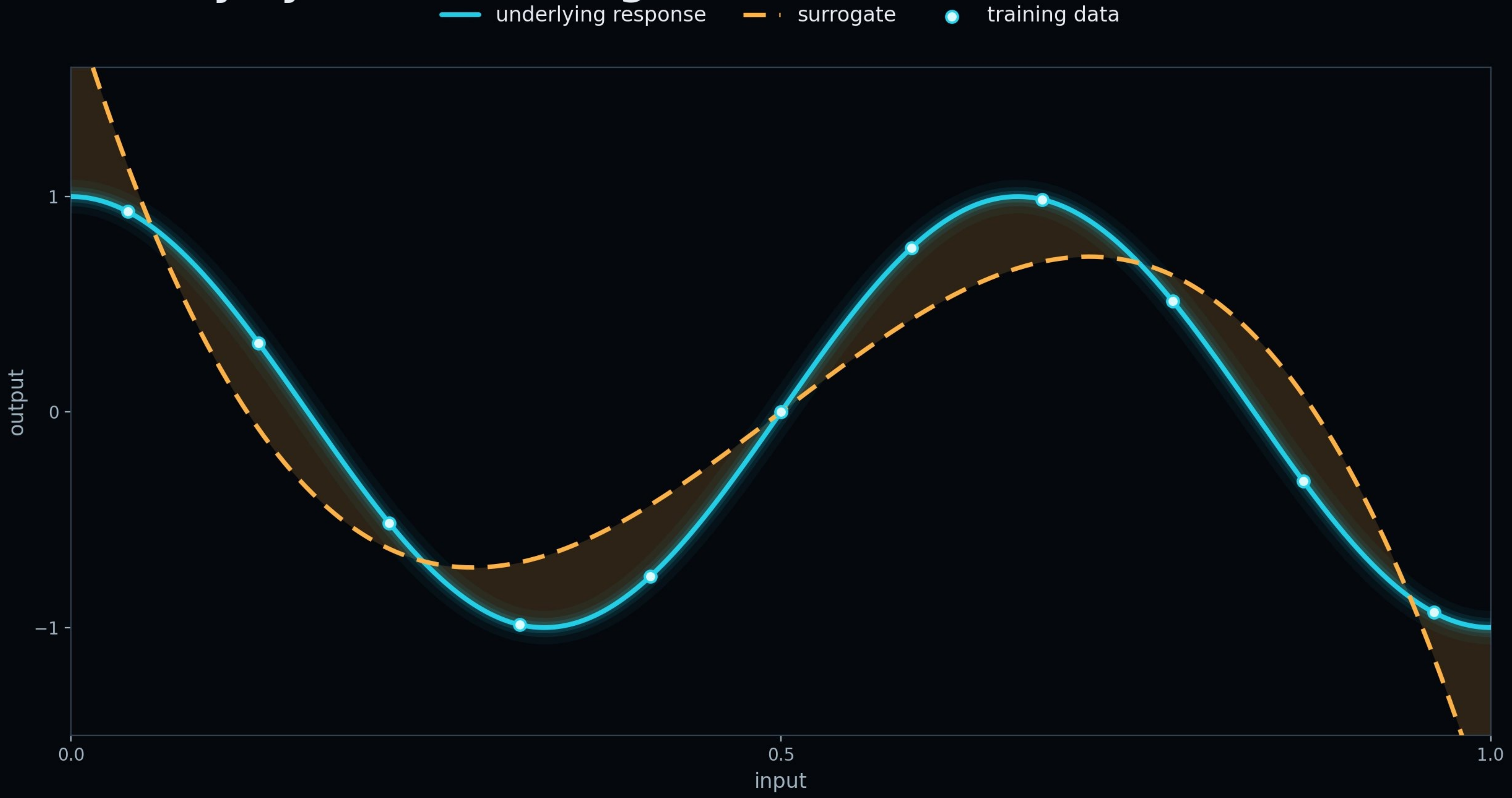
EXPERIMENT

physical test

SIMULATION

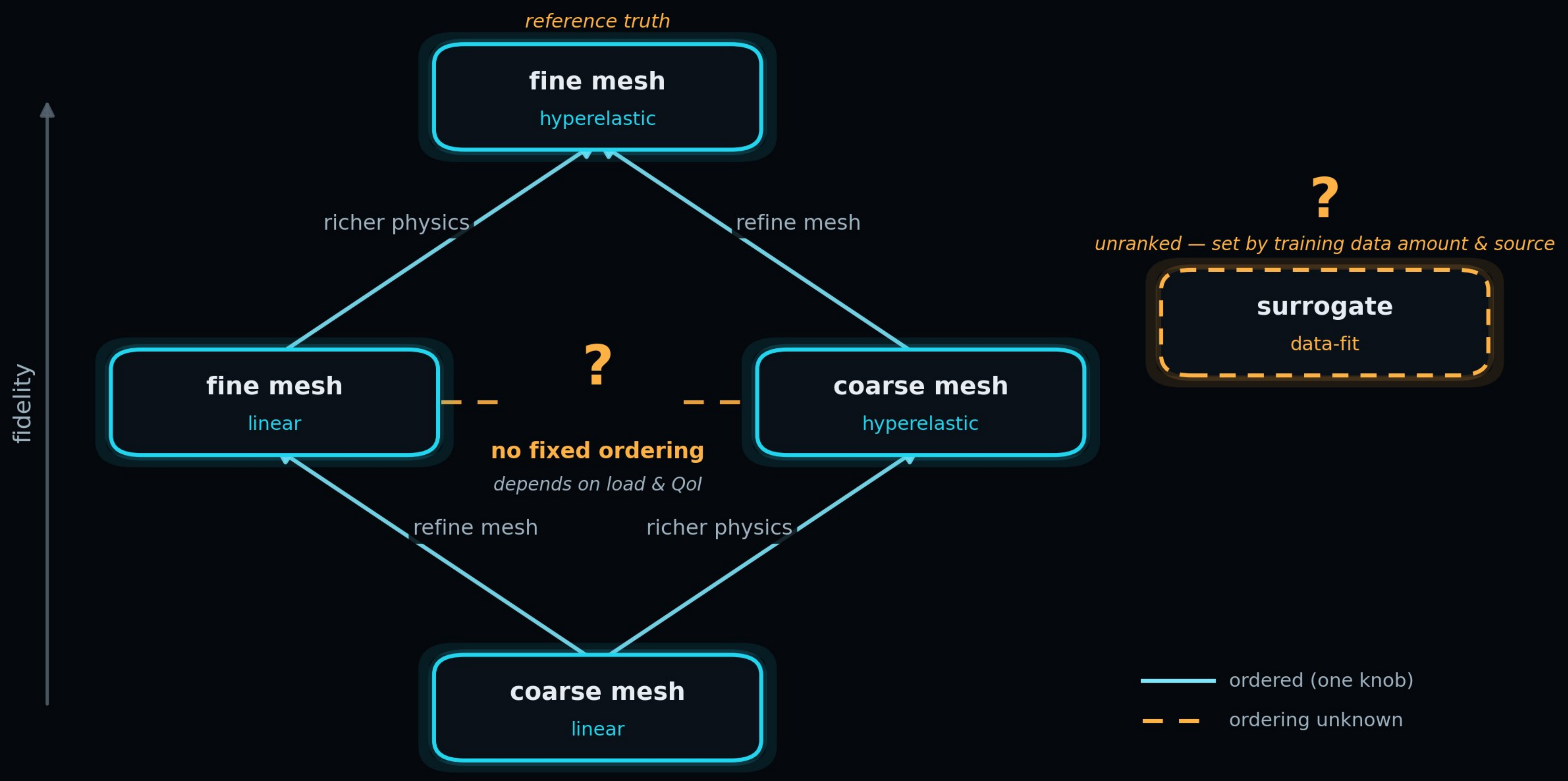
digital twin

Fidelity by data-fit surrogate



Fidelity is not a scalar

some models are ordered; others are not — the ensemble forms a partial order, measured from a pilot

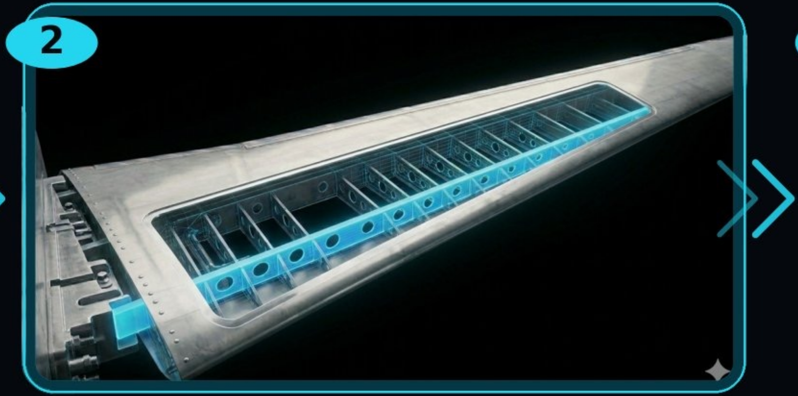


Fidelity by model scope

the same physics, modeled at shrinking extent — from the certified vehicle to a cheap benchmark

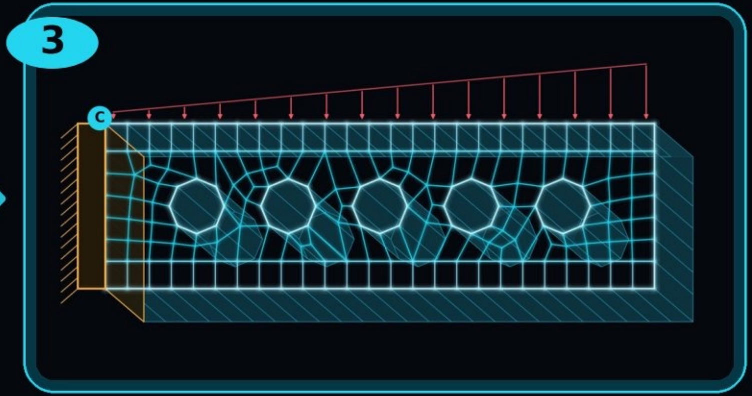
AIRCRAFT

the real problem



WING BOX

a load-carrying component



BEAM PANEL

our benchmark model

02

SECTION 02 OF 06

Monte Carlo

The most general estimator: its rate ignores dimension and smoothness — but that rate is slow.

PyApprox tutorials: Monte Carlo Sampling · Estimator Accuracy & MSE · MC Budget Estimation

Slides • 8 min

The Monte Carlo Estimator

$$\hat{\mu} = \frac{1}{N} \sum_{i=1}^{N} f(\theta^{(i)})$$

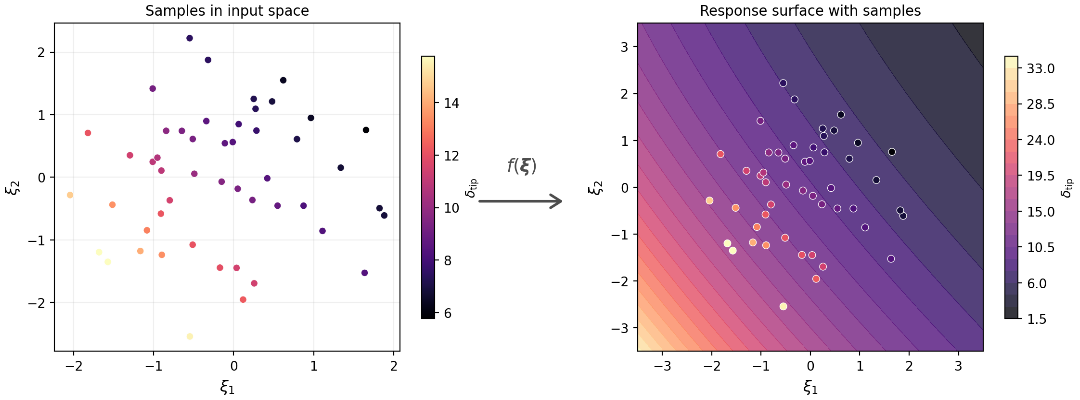
$$S = \{\theta^{(1)}, \ldots, \theta^{(N)}\}, \quad \theta^{(i)} \sim p(\theta)$$

Dimension-free. The convergence rate is the same in 2 or 200 input dimensions.

Smoothness-free. The convergence rate does not depend on smoothness of f

Challenge. Error shrinks only as fast as the square root of the sample count — halving it takes four times as many samples.

From samples to the response surface



Left: input samples colored by tip deflection. Right: the model's response surface — the input→output map MC averages over.

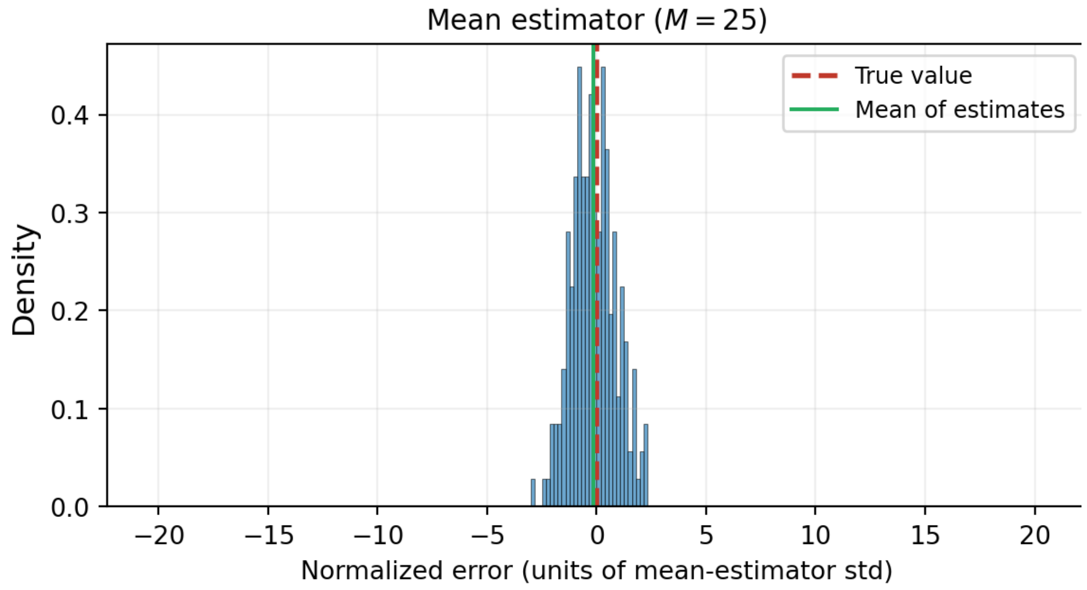
Run It Twice, Get Two Answers

Each MC run draws fresh samples, so the estimate is itself a random variable — repeat it and the values scatter.

Centered = unbiased. The estimates cluster on the true value; their average is the true mean.

Spread = variance. How widely they scatter is the estimator’s variance — exactly what sets MC accuracy.

Estimates scatter, run to run



Repeated MC experiments (M = 25). Estimates cluster tightly and center on the truth — red and green coincide, so the mean estimator is unbiased.

Accuracy Is Just Variance

MEAN-SQUARED ERROR

$$\mathrm{MSE}[\hat{\mu}] = \mathbb{E}_S\left[(\hat{\mu} - \mu)^2\right] = (\mathbb{E}_S[\hat{\mu}] - \mu)^2 + \mathbb{V}_S[\hat{\mu}]$$

$$\mathbb{E}_S[\hat{\mu}] = \mu \Rightarrow \mathrm{MSE}[\hat{\mu}] = \mathbb{V}_S[\hat{\mu}] = \frac{\sigma^2}{N}$$

MC is unbiased, so accuracy is set entirely by estimator variance — the lever every multi-fidelity method pulls.

The Slow Root-N Rate

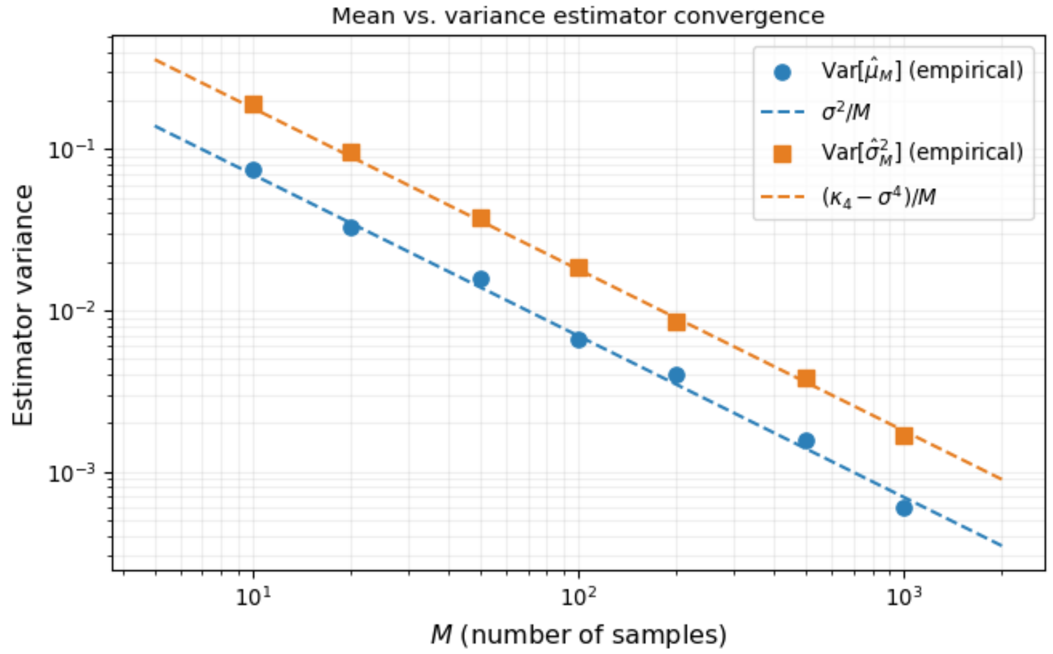
Estimator variance falls in proportion to the sample count — so the error, its square root, falls only as fast as the square root of the count.

The root-N wall. Cutting the error tenfold costs a hundredfold more samples; a hundredfold costs ten-thousandfold.

Variance is harder. The variance estimator obeys the same rate but with a larger constant, so it always sits above the mean.

For an expensive high-fidelity model, this is the wall multi-fidelity methods break.

Error decays at the root-N rate

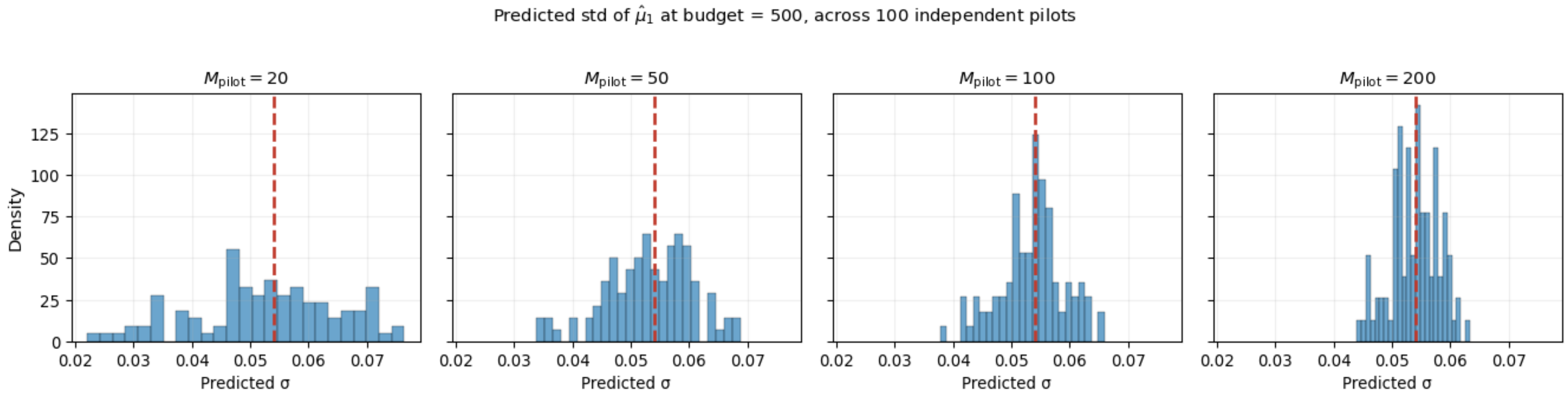


both fall in proportion to the count, and the variance estimator sits on the higher line.

Sizing the Budget Is Circular

$$N \gtrsim \frac{\mathbb{V}_\theta[f]}{\varepsilon^2} = \frac{\sigma^2}{\varepsilon^2}$$

The budget is set by the QoI variance — exactly what MC is trying to estimate. You size it from a pilot run, so the budget is only as good as the pilot; for an expensive model the required N is unaffordable.



Predicted budget across 100 independent pilots (truth dashed); reliability grows with pilot size — cheap correlated models break the loop.

03

SECTION 03 OF 06

Two Models: Control Variates → ACV

Borrow a cheap correlated model to cut variance, then drop the known-mean assumption.

PyApprox tutorials: Control Variate Monte Carlo · ACV Concept

Slides • 14 min

Why Not Just Use the Cheap Model?

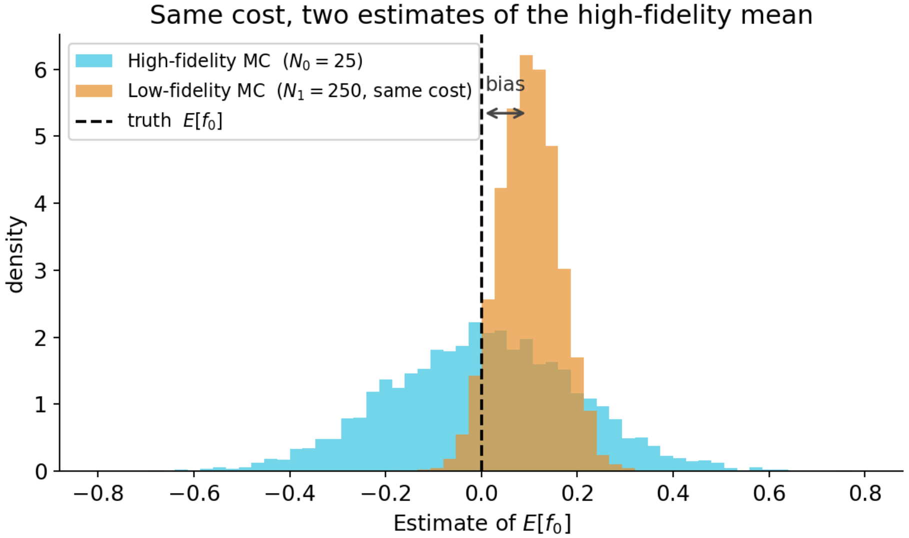
PRECISE BUT BIASED

For a fixed budget you can run the costly model a few times — or the cheap model many times. Neither estimate is good enough on its own.

High fidelity. Unbiased — but the budget buys few runs, so it scatters widely.

Low fidelity. Cheap, so the budget buys many runs: tight, but centered on the wrong value.

Control variates. Keep the cheap precision, cancel the bias using correlation.



Same cost (HF cost 1, LF cost 0.1 — so 10× more cheap runs). The high-fidelity estimate centers on the truth but scatters; the low-fidelity estimate is far tighter but offset by its model bias. PyApprox tunable ensemble, correlation 0.97.

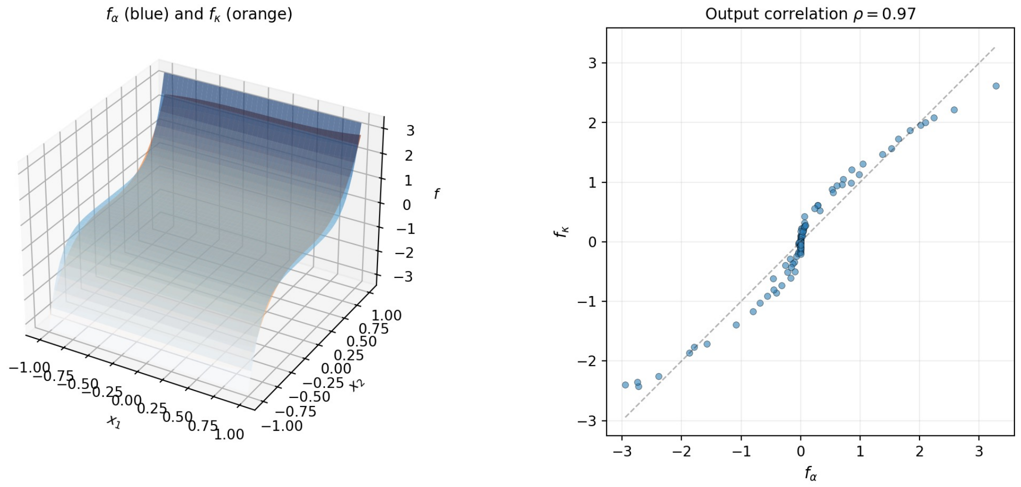
Borrow a Cheap, Correlated Model

Suppose a cheap low-fidelity model tracks the expensive one — when one output is large, so is the other.

Known mean. If the cheap model's mean is known, its MC error is a free, mean-zero signal.

Use it to cancel. Subtract a scaled copy of that error from the HF estimate to shrink its variance.

High- and low-fidelity, tightly correlated



The high- and low-fidelity models share shape; the scatter shows an output correlation of about 0.97.

The Control-Variate Estimator

UNBIASED FOR ANY WEIGHT

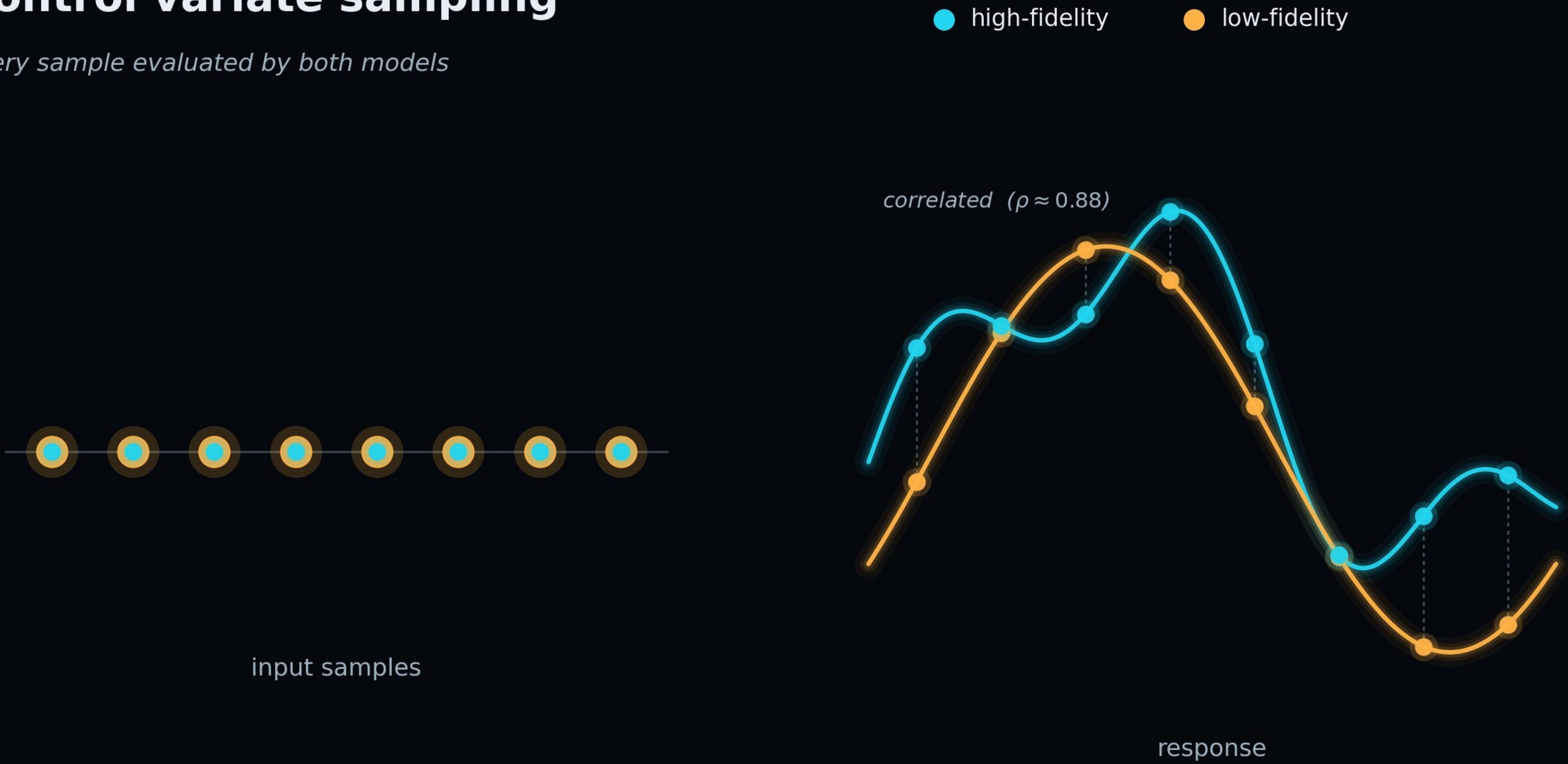
$$\hat{\mu}_{\alpha}^{\mathrm{CV}} = \hat{\mu}_{\alpha}(\mathcal{Z}_N) + \eta\left(\hat{\mu}_{\kappa}(\mathcal{Z}_N) - \mu_{\kappa}\right)$$

$$\mathbb{E}_S[\hat{\mu}_{\alpha}^{\mathrm{CV}}] = \mu_{\alpha}$$

The correction has mean zero, so adding it introduces no bias — the weight is free to minimize variance.

Control variate sampling

every sample evaluated by both models



Choosing the Optimal Weight

VARIANCE IS A QUADRATIC IN THE WEIGHT

$$\mathbb{V}_S[\hat{\mu}_\alpha^{\mathrm{CV}}] = \mathbb{V}_S[\hat{\mu}_\alpha] + \eta^2 \mathbb{V}_S[\hat{\mu}_\kappa] + 2\eta \mathrm{Cov}_S(\hat{\mu}_\alpha, \hat{\mu}_\kappa)$$

minimize over the weight (set the derivative to zero):

$$\eta^* = -\frac{C_{\alpha\kappa}}{\sigma_\kappa^2} = -\frac{\mathrm{Cov}_\theta(f_\alpha, f_\kappa)}{\mathbb{V}_\theta[f_\kappa]}$$

The optimal weight is the regression coefficient of the HF error on the LF error — it minimizes the estimator variance.

Variance Reduction Depends Only on Correlation

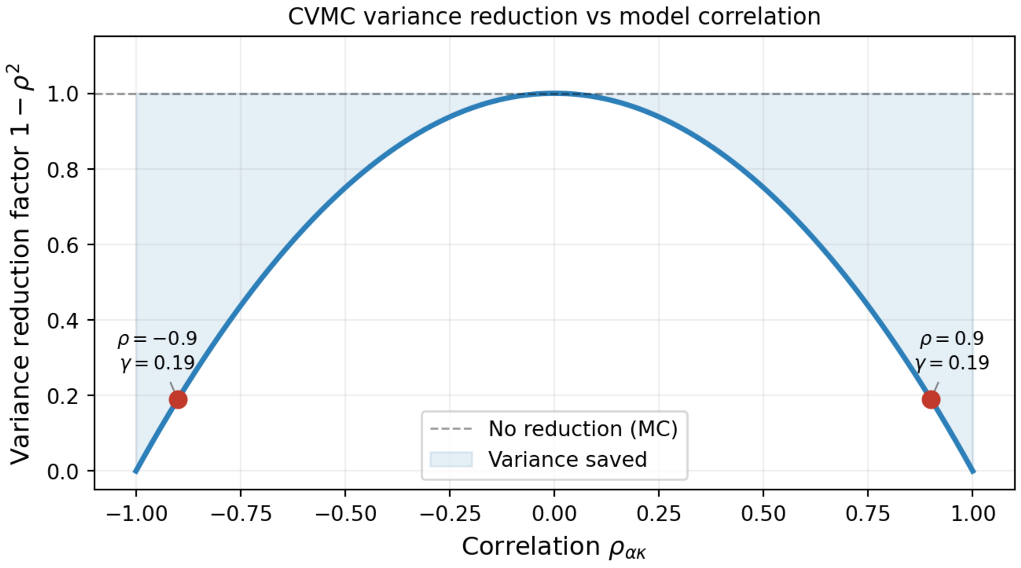
$$\mathbb{V}_S[\hat{\mu}_\alpha^{\mathrm{CV}}] = (1 - \rho_{\alpha\kappa}^2)\mathbb{V}_S[\hat{\mu}_\alpha]$$

$$\rho_{\alpha\kappa} = \mathrm{Corr}(f_\alpha, f_\kappa) \in [-1, 1]$$

The reduction factor is fixed by the model correlation alone — not by sample size or model variance.

Strong correlation wins. A correlation of 0.9 cuts variance by 81%; 0.99 cuts it by 99%; near-zero correlation gives nothing.

Reduction depends only on correlation



The reduction factor versus model correlation.

Control Variates in Practice

For the same target cost, the CVMC estimator scatters far less, and stays centered on the truth.

The catch. CVMC needs the cheap model's mean known exactly. When it isn't, you estimate it too — the idea behind Approximate Control Variates.

Same cost, far less spread

100 samples, correlation 0.97, 1000 trials. Both unbiased; CVMC variance is about 5% of MC.

CVMC: Budget Pins Sample Count

THE GENERAL PROBLEM

$$\min_{\theta} \mathbb{V}[\hat{\mu}(\theta)] \quad \text{s.t.} \quad \text{Cost}(\theta) \leq P$$

Same N for both models

$$P = N\,(c_\alpha + c_\kappa)$$

Weight is covariance-set — independent of N

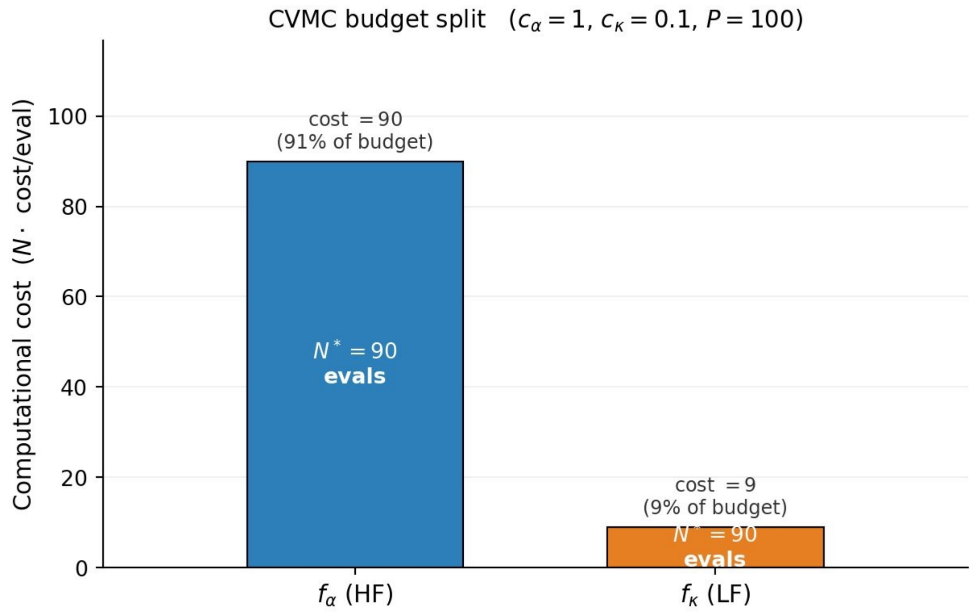
$$\eta^\star = -C_{\alpha\kappa}/\sigma_\kappa^2$$

So, the budget alone fixes the count

$$N^\star = \lfloor P/(c_\alpha + c_\kappa) \rfloor$$

A closed form — no optimization.

Where the budget goes

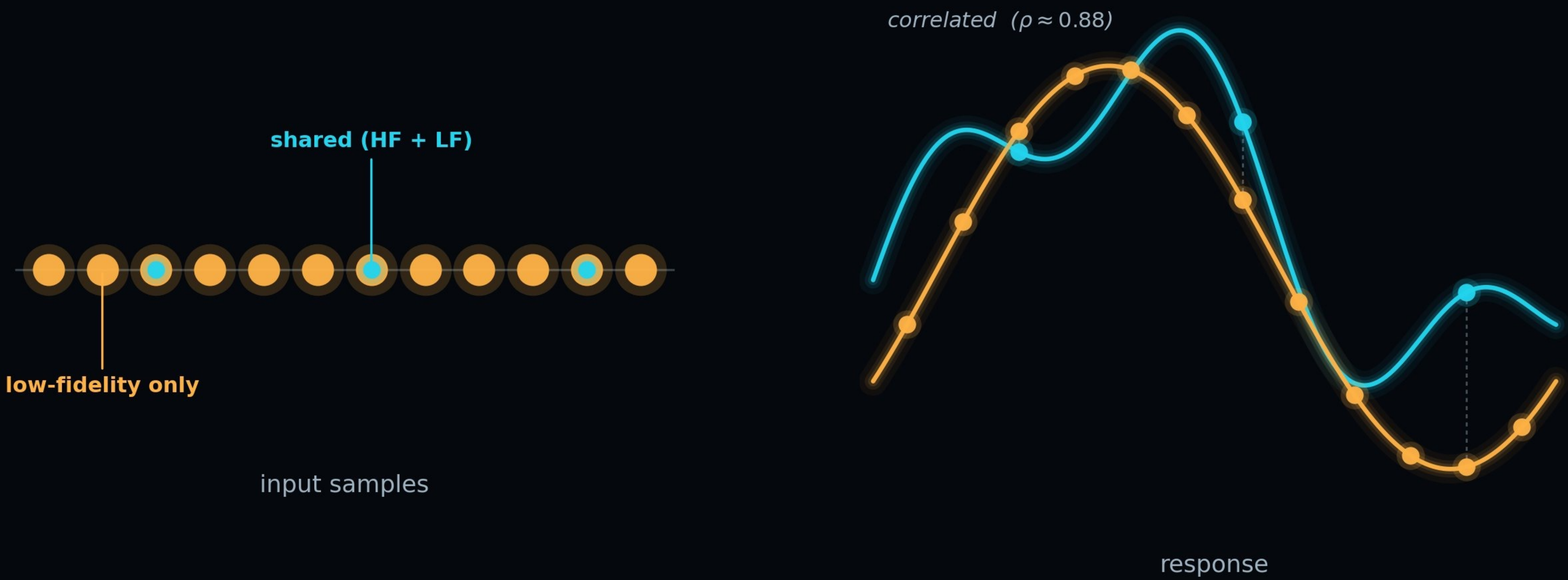


Both models run the same N = 90, yet the costly model claims 91% of the budget..*

ACV sampling

a few shared samples + many low-fidelity-only samples

● high-fidelity ● low-fidelity



shared (HF + LF)

low-fidelity only

input samples

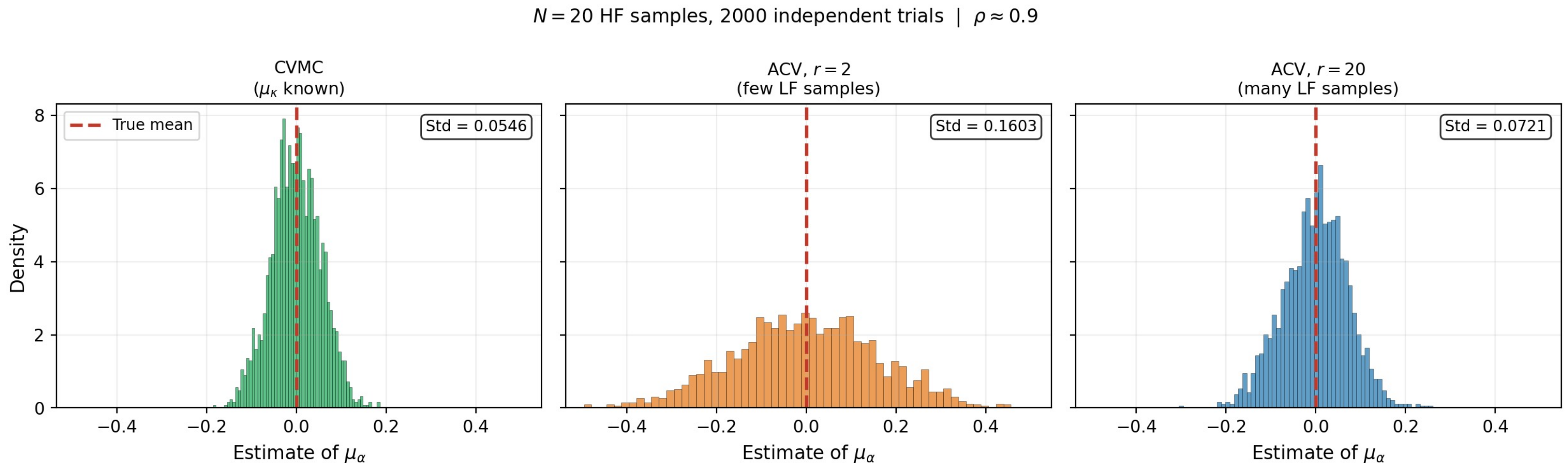
correlated (ρ ≈ 0.88)

response

When the Cheap Mean Is Unknown

UNBIASED FOR ANY r > 1 — THE LF MEAN NEVER APPEARS

$$\hat{\mu}_{\alpha}^{\mathrm{ACV}} = \hat{\mu}_{\alpha}(\mathcal{Z}_N) + \eta\left(\hat{\mu}_{\kappa}(\mathcal{Z}_N) - \hat{\mu}_{\kappa}(\mathcal{Z}_{rN})\right)$$



Distribution of the estimate over 2000 trials (20 HF samples, correlation about 0.9): with the LF mean known (left) it is tightest; a small LF sample (r = 2, center) is noisy; a large one (r = 20, right) nearly recovers it.

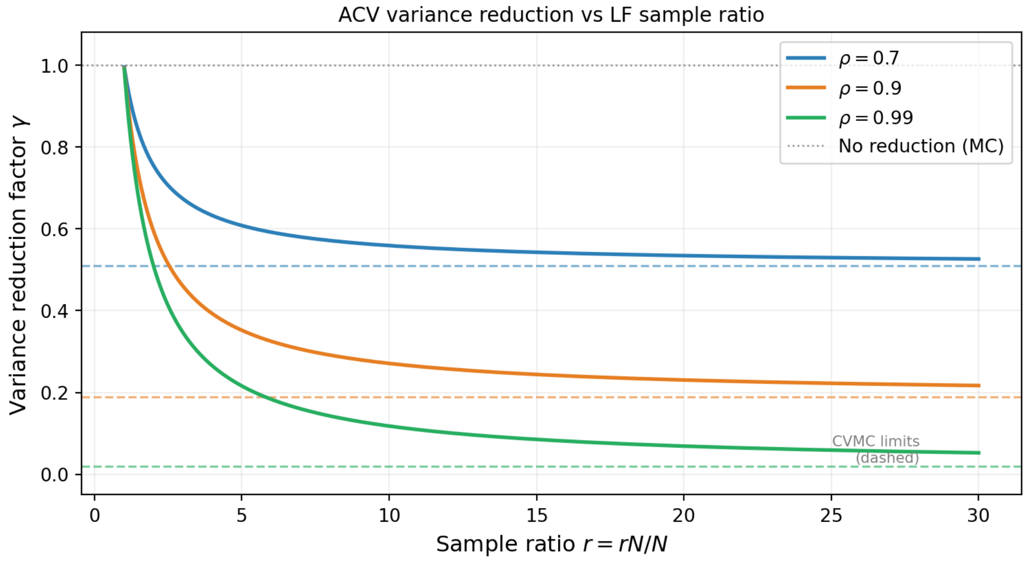
More Cheap Samples, More Reduction

$$\gamma = 1 - \frac{r-1}{r}\rho_{\alpha\kappa}^2$$

The reduction factor interpolates between two limits:

Few LF samples: no reduction — you recover plain Monte Carlo.

Many LF samples: the full CVMC reduction floor is returned. Because the LF model is cheap, a large ratio is affordable — about ten already captures roughly 90% of the gain.



The reduction factor versus the LF-to-HF sample ratio, for three correlations. Dashed lines mark each CVMC limit.

04

SECTION 04 OF 06

Many Models: GroupACV

The unifier — one optimal-weight estimator subsuming MLMC, MFMC, and MLBLUE.

PyApprox tutorials: General ACV · Group ACV Concept · MLBLUE

Slides • 14 min

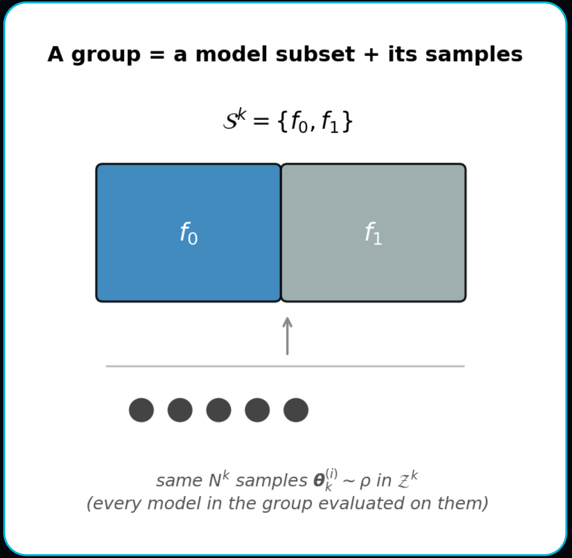
Every method is a Group ACV Estimator

YOU PICK THE GROUPS — THE FRAMEWORK PICKS THE WEIGHTS

$$\hat{Q}_{\mathrm{GACV}} = \sum_{k=1}^{K} (\beta^k)^\top \hat{Q}^k(\mathcal{Z}^k)$$

$$\hat{Q}^k(\mathcal{Z}^k) = \left(\hat{Q}_\ell(\mathcal{Z}^k)\right)_{\ell \in \mathcal{S}^k}$$

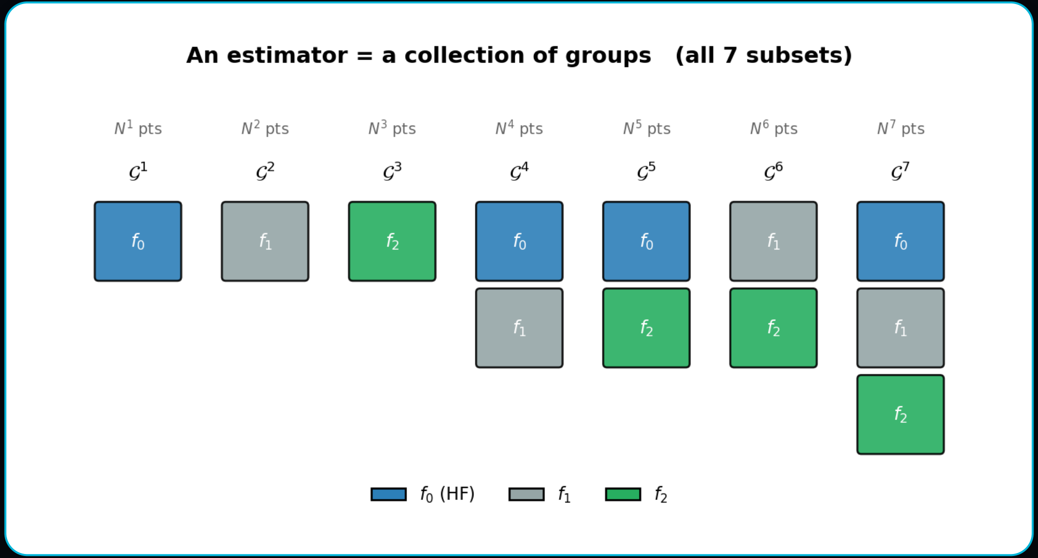
Each entry estimates a statistic Q from the group's shared samples — the mean by default, with variance or mean-and-variance in the general case.



A group = a model subset + its samples

$\mathcal{S}^k = \{f_0, f_1\}$

same N^k samples $\boldsymbol{\theta}_k^{(i)} \sim \rho$ in $\mathcal{Z}^k$
(every model in the group evaluated on them)



An estimator = a collection of groups (all 7 subsets)

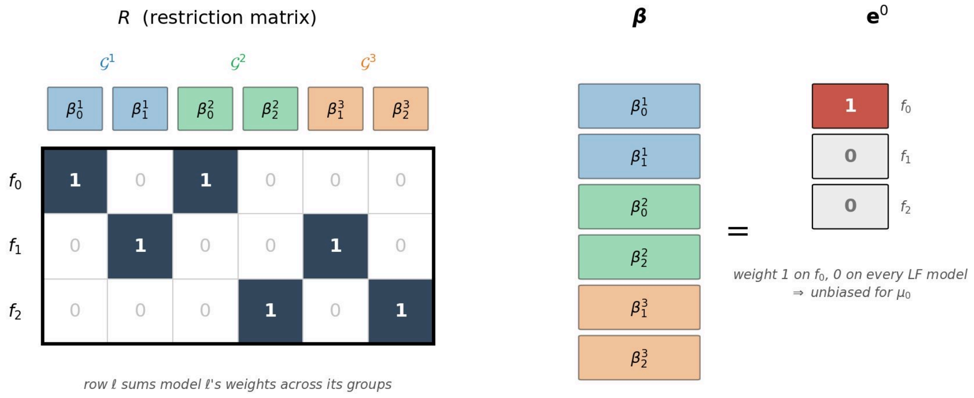
One Constraint, One Formula

UNBIASEDNESS FIXES THE WEIGHTS

$$R\beta = e^0$$

$$\beta^\star = C^{-1}R^\top(RC^{-1}R^\top)^{-1}e^0$$

$$\mathbb{V}[\hat{Q}_{\mathrm{GACV}}] = (\mathbf{e}^0)^\top(R\,C^{-1}R^\top)^{-1}\mathbf{e}^0$$

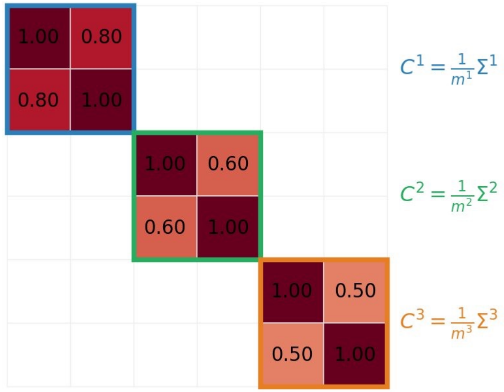


$$\begin{array}{c|cccccc} & \beta_0^1 & \beta_1^1 & \beta_0^2 & \beta_2^2 & \beta_1^3 & \beta_2^3 \\ \hline f_0 & 1 & 0 & 1 & 0 & 0 & 0 \\ f_1 & 0 & 1 & 0 & 0 & 1 & 0 \\ f_2 & 0 & 0 & 0 & 1 & 0 & 1 \end{array} \begin{bmatrix} \beta_0^1 \\ \beta_1^1 \\ \beta_0^2 \\ \beta_2^2 \\ \beta_1^3 \\ \beta_2^3 \end{bmatrix} = \begin{array}{cc} 1 & f_0 \\ 0 & f_1 \\ 0 & f_2 \end{array}$$

row ℓ sums model ℓ's weights across its groups

weight 1 on f_0, 0 on every LF model ⇒ unbiased for μ_0

Restriction matrix R: each row sums a model's weights across its groups; the target forces unit weight on the high-fidelity model and zero on the rest.



$$\begin{bmatrix} 1.00 & 0.80 & & & & \\ 0.80 & 1.00 & & & & \\ & & 1.00 & 0.60 & & \\ & & 0.60 & 1.00 & & \\ & & & & 1.00 & 0.50 \\ & & & & 0.50 & 1.00 \end{bmatrix}$$

$C^1 = \frac{1}{m^1}\Sigma^1$

$C^2 = \frac{1}{m^2}\Sigma^2$

$C^3 = \frac{1}{m^3}\Sigma^3$

Per-group covariance is block-diagonal — groups draw independent samples, so off-diagonal blocks vanish.

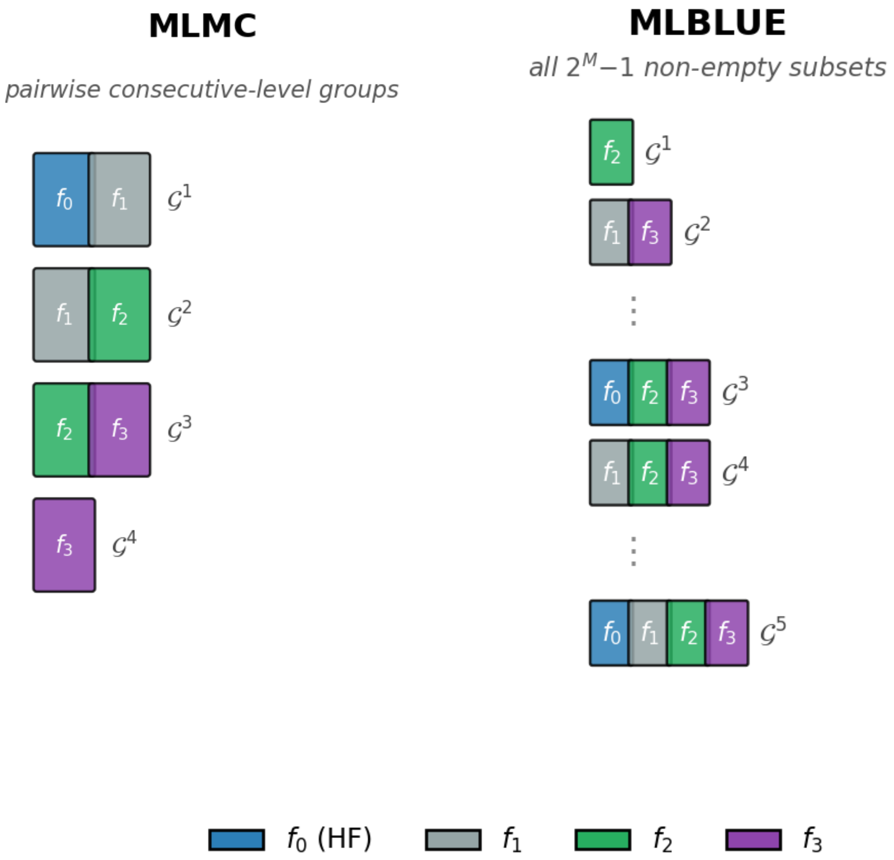
MLMC, MFMC, ACVMF, MLBLUE Are All Groups

Each familiar estimator is just a choice of group structure:

Pairwise / nested. MLMC and MFMC chain consecutive levels.

ACVMF. Every LF model attaches directly to the high-fidelity model, independent of the others — a flat two-level tree, not a hierarchy.

All subsets. MLBLUE explores the full design space. Same optimal-weight formula throughout — only the constraint and covariance change.



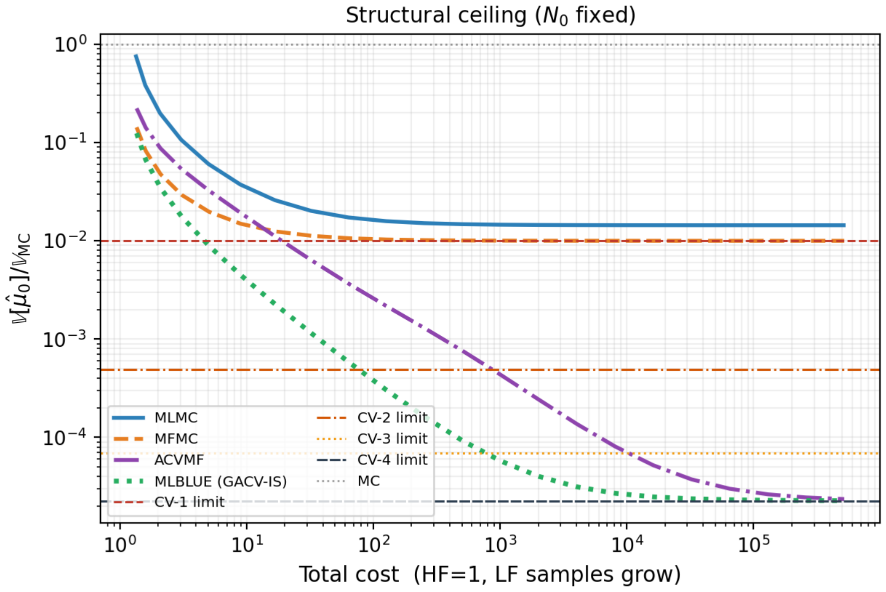
Each bar is one group; colored blocks show which models share that group's samples.

Structure Sets the Ceiling

How low can each estimator go? With the HF count fixed, the group structure sets a hard floor.

CV-1 ceiling. Pairwise & nested (MLMC, MFMC) stop here.

CV-M ceiling. ACVMFand MLBLUE go far lower.



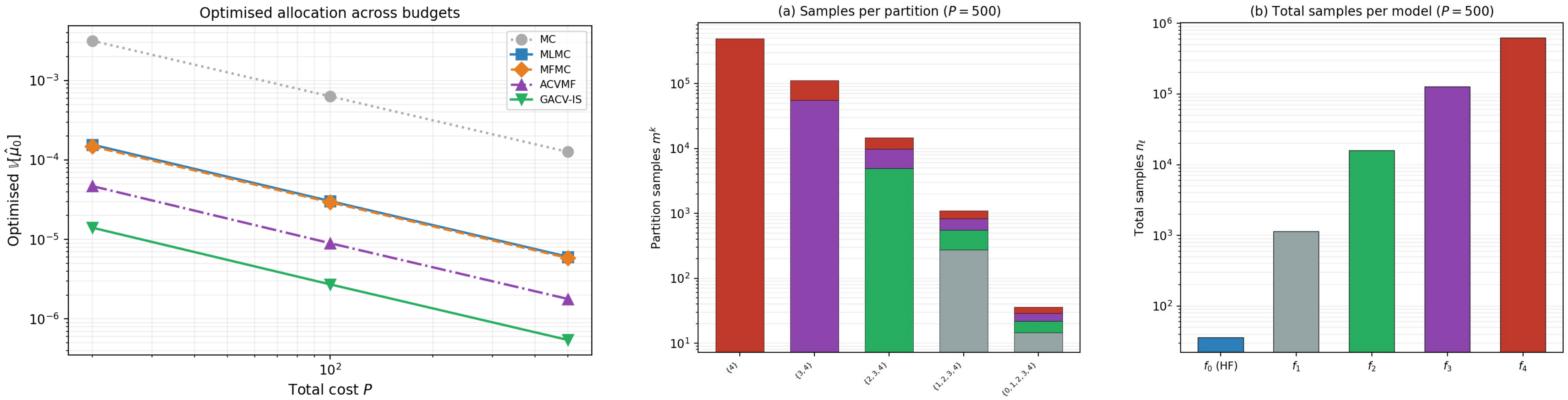
With the high-fidelity sample count fixed, MLMC and MFMC plateau at the CV-1 ceiling while ACVMF and MLBLUE reach the far lower CV-M floor.

Solving the Allocation Problem

OPTIMIZE THE COUNTS, NOT THE WEIGHTS

Unbiasedness already fixes the weights for any allocation — so GroupACV's only freedom is the per-group sample counts.

$$\min_{m^k} \mathbb{V}[\hat{Q}_{\mathrm{GACV}}] \quad \mathrm{s.t.} \quad \sum_k m^k c^k \le P \ \textit{with} \qquad c^k = \sum_{\ell \in S^k} c_\ell$$



Left: optimized variance keeps dropping with budget. Middle and right: of 31 possible subsets only about five carry samples — the budget lands almost entirely on the cheapest model ($f_4 \approx 6\times10^5$) while the high-fidelity model is barely sampled ($f_0 \approx 30$).

Three Extensions for Free

One group collection, one constraint — three generalizations, with no new machinery:

Multiple QoIs
Estimate several outputs jointly — cross-QoI covariance within each group becomes free signal.

Statistics beyond the mean
Variance, or mean-and-variance, through the same optimal-weight result; only the per-group covariance changes.

Mixed known statistics
Drop the constraint rows for models with analytic moments — interpolating between full ACV and full CVMC.

None of these change the framework — they are why it exists.

05

SECTION 05 OF 06

Extensions & the Cookbook

Beyond the mean, known means, pilot studies, and the sample-allocation problem.

PyApprox tutorials: Group ACV Multi-Statistic · Pilot Studies · API Cookbook

Slides • 10 min

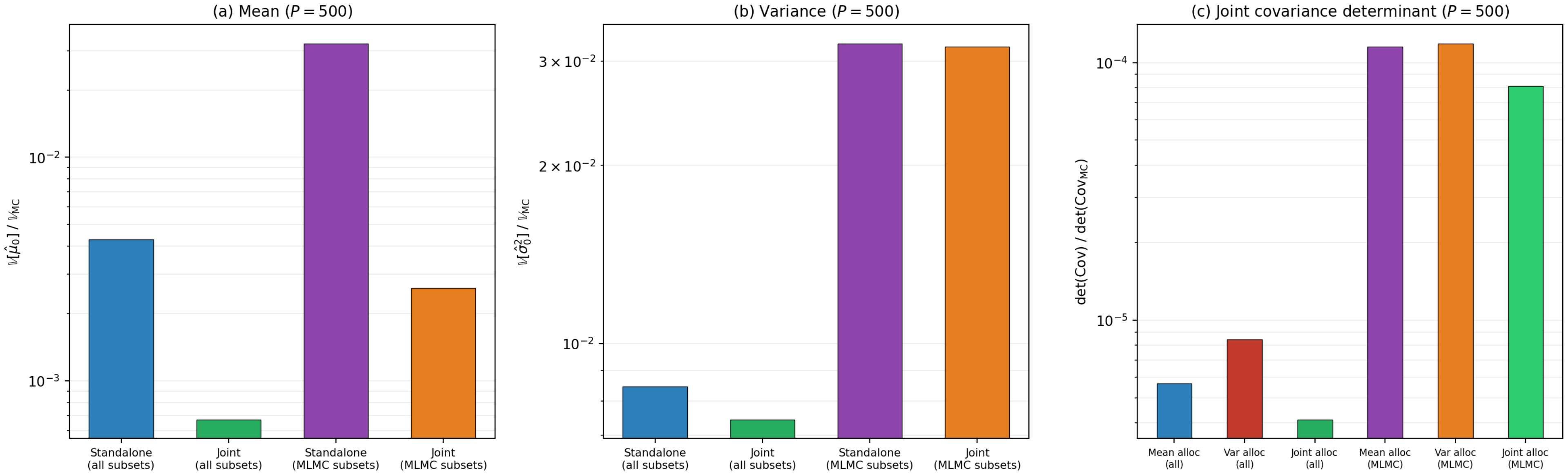
Statistics Beyond the Mean

The same groups, the same constraint, the same optimal weights — only the per-group covariance changes.

Mean: the per-group covariance.

Variance: adds a fourth-order moment.

Mean + variance: adds a mean–variance cross term.



Polynomial 5-model, P = 500. Estimating mean and variance JOINTLY lowers the variance of both vs estimating each alone — a mean–variance cross term couples them, the multi-output gain applied across statistics.

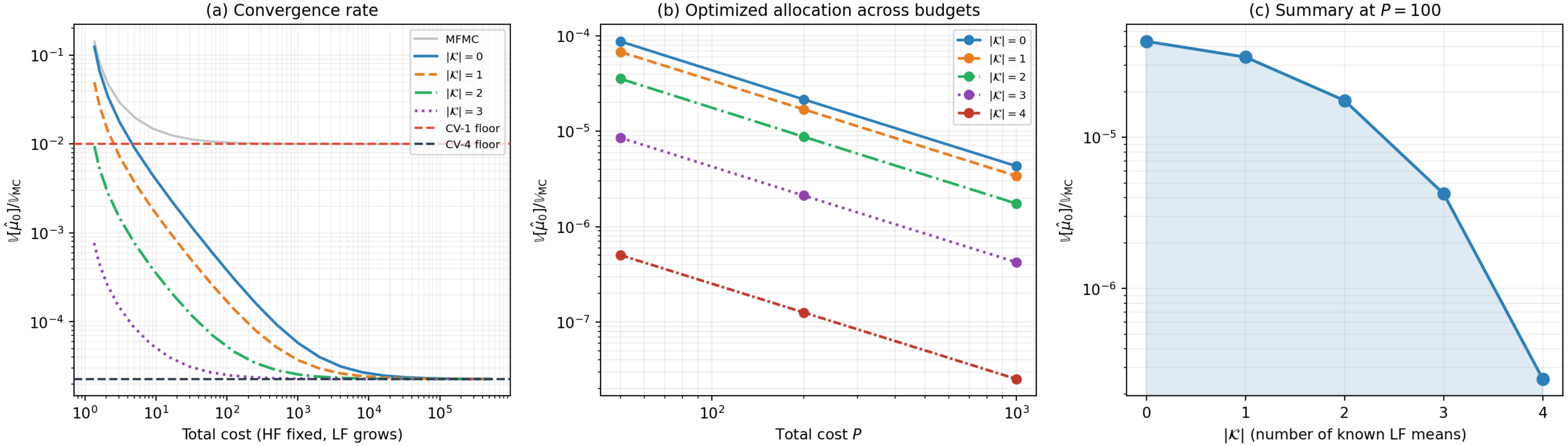
Known Means: From ACV Toward CVMC

A known low-fidelity mean removes one unbiasedness constraint — strictly enlarging the feasible weights, strictly lowering variance.

Walk the spectrum: from full ACV (no means known) to full CVMC (all means known).

Same CV-M floor — reached sooner.

Sources: PCE, GP posteriors, analytic baselines.



Poly 5-model, known means added cheapest-first. Left: convergence to the CV-M floor for zero to three known means. Right: variance versus total cost — more known means, lower variance at every budget.

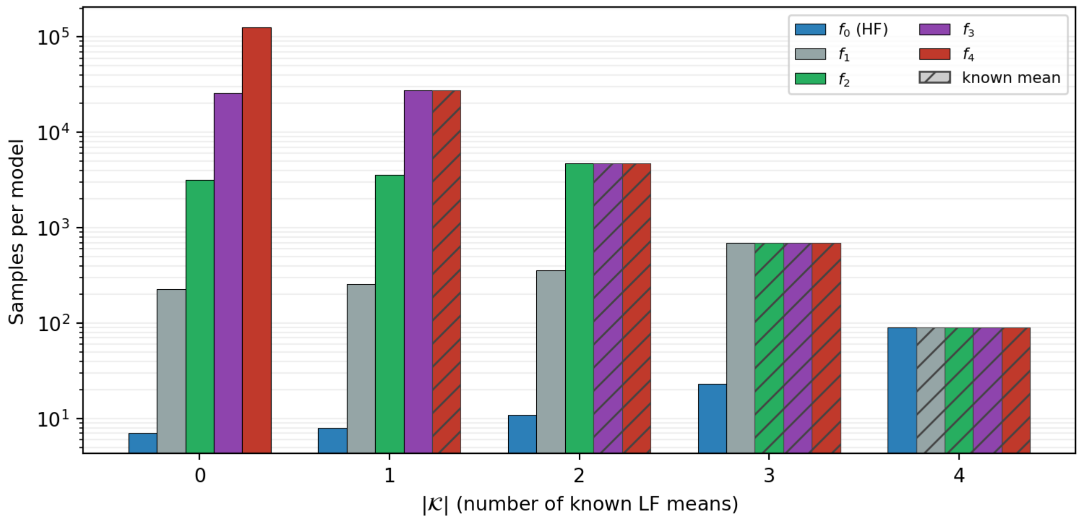
Where Known Means Send the Budget

Known means reshape the optimal allocation:

One partition drops per known mean — the subset whose models are all now known disappears.

The HF model gains samples — freed budget flows to the remaning model s(about thirteen times more HF samples as known means grow here).

Known models stop being over-sampled — they equalize with the cheapest unknown model.

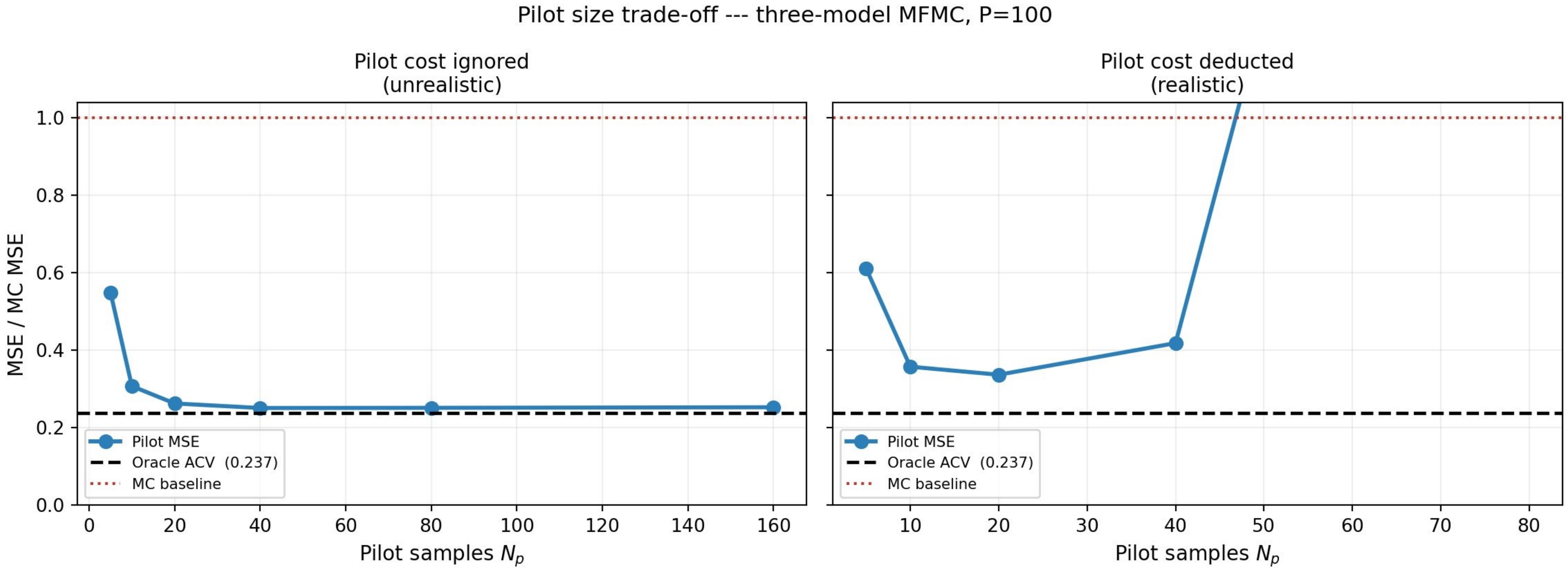


Per-model allocation at P = 100 as the number of known means grows from none to four (hatched bars mark known means).

The Pilot Study

Both the allocation and the optimal weights need the model covariance — but estimating it means running the models and spending budget. That circular dependency is the pilot study.

MC vs ACV: MC needs only one model's variance to size its sample count; ACV and GroupACV need the full cross-covariance — so every model is run at the same shared pilot points.



MSE versus pilot size. Left: when the pilot is free, more samples only help — error falls to the estimator variance. Right: charged to the budget, too small is noisy and too large starves the estimator — an optimal size sits between. (Shown for a representative multi-fidelity estimator; the same trade-off holds for all GroupACV estimators.)

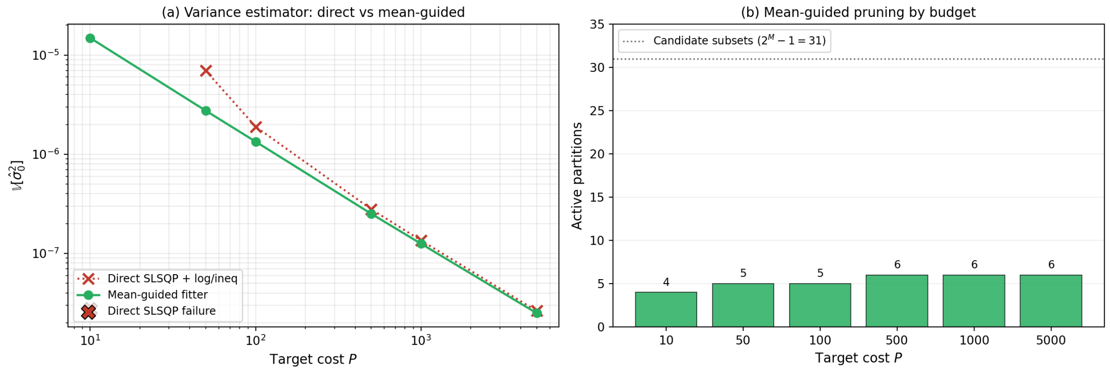
Solving the Allocation Problem

Searching is free. Allocation optimizes over the pilot covariance — comparing estimators costs optimizer iterations, never new simulations.

The default is robust. SLSQP with log-scaling and an inequality budget works out of the box.

Roll your own with care. Naïve scaling degrades at large budgets; the convex SDP looks appealing but is numerically unreliable on realistic problems.

Variance needs at least two samples per group. A cheap mean-guided pre-screen finds the active groups before the variance solve.



Variance estimation: the mean-guided fitter stays feasible and beats a direct solve at every budget; right, how aggressively it prunes.

06

SECTION 06 OF 06

Results, Pitfalls & Wrap-Up

The wing-spar case study, the research frontiers, and when to reach for this.

PyApprox tutorials: Aerospace Beam Case Study · Ensemble Selection

Slides • 8 min

Case Study: A Wing-Spar Panel

Certifying the panel needs the statistics of two reliability QoIs — tip deflection and von Mises stress — but the high-fidelity model is far too costly to sample directly.

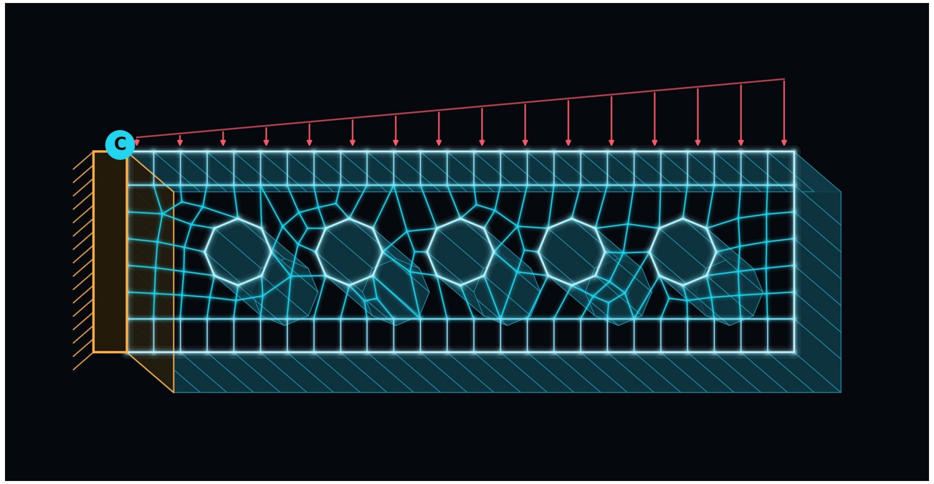
Nonlinear, fine mesh — the HF reference (cost 1).

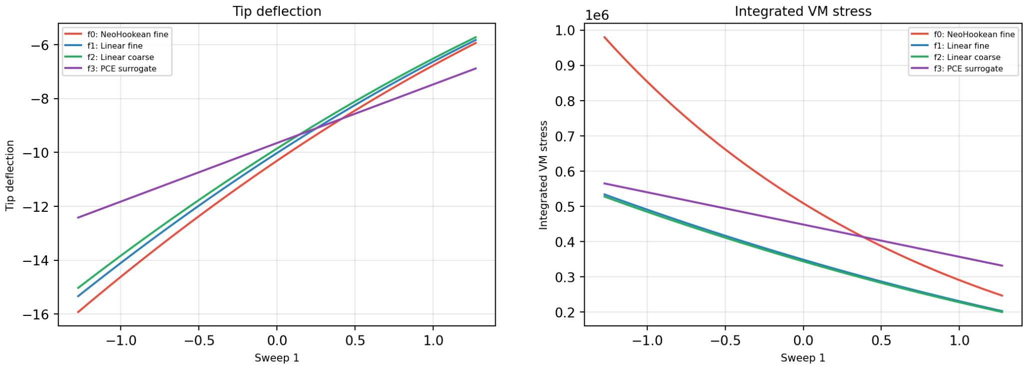
Linear, same mesh — drops the physics (about 0.3).

Linear, coarse mesh — drops physics and resolution (about 0.02).

PCE surrogate — data-fit, with a known mean (about 1e-5).

Shared random inputs couple them, but they cannot be ordered : a clean refinement pair, a load-dependent pair, and a surrogate not comparable at all — the case GroupACV handles and MLMC cannot.



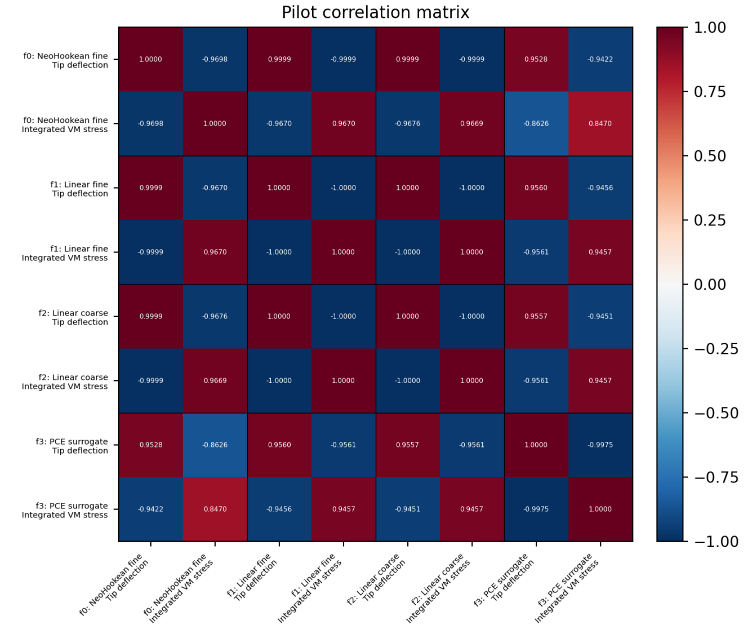


1-D sweeps: low-fidelity models are biased relative to the HF, yet their trends track it — the consistent correlation multi-fidelity UQ exploits.

What the Pilot Reveals

A 50-sample pilot evaluates every model at the same inputs — everything the allocation needs comes from it.

The correlations expose the order. Linear models are near-perfect for deflection (about 0.9999) and strong for stress (about 0.97); the cheap surrogate is high for deflection (0.95) but markedly weaker for stress (0.85).



Pilot correlation matrix

	f0: NeoHookean fine Tip deflection	f0: NeoHookean fine Integrated VM stress	f1: Linear fine Tip deflection	f1: Linear fine Integrated VM stress	f2: Linear coarse Tip deflection	f2: Linear coarse Integrated VM stress	f3: PCE surrogate Tip deflection	f3: PCE surrogate Integrated VM stress
f0: NeoHookean fine Tip deflection	1.0000	-0.9698	0.9999	-0.9999	0.9999	-0.9999	0.9528	-0.9422
f0: NeoHookean fine Integrated VM stress	-0.9698	1.0000	-0.9670	0.9670	-0.9676	0.9669	-0.8626	0.8470
f1: Linear fine Tip deflection	0.9999	-0.9670	1.0000	-1.0000	1.0000	-1.0000	0.9560	-0.9456
f1: Linear fine Integrated VM stress	-0.9999	0.9670	-1.0000	1.0000	-1.0000	1.0000	-0.9561	0.9457
f2: Linear coarse Tip deflection	0.9999	-0.9676	1.0000	-1.0000	1.0000	-1.0000	0.9557	-0.9451
f2: Linear coarse Integrated VM stress	-0.9999	0.9669	-1.0000	1.0000	-1.0000	1.0000	-0.9561	0.9457
f3: PCE surrogate Tip deflection	0.9528	-0.8626	0.9560	-0.9561	0.9557	-0.9561	1.0000	-0.9975
f3: PCE surrogate Integrated VM stress	-0.9422	0.8470	-0.9456	0.9457	-0.9451	0.9457	-0.9975	1.0000

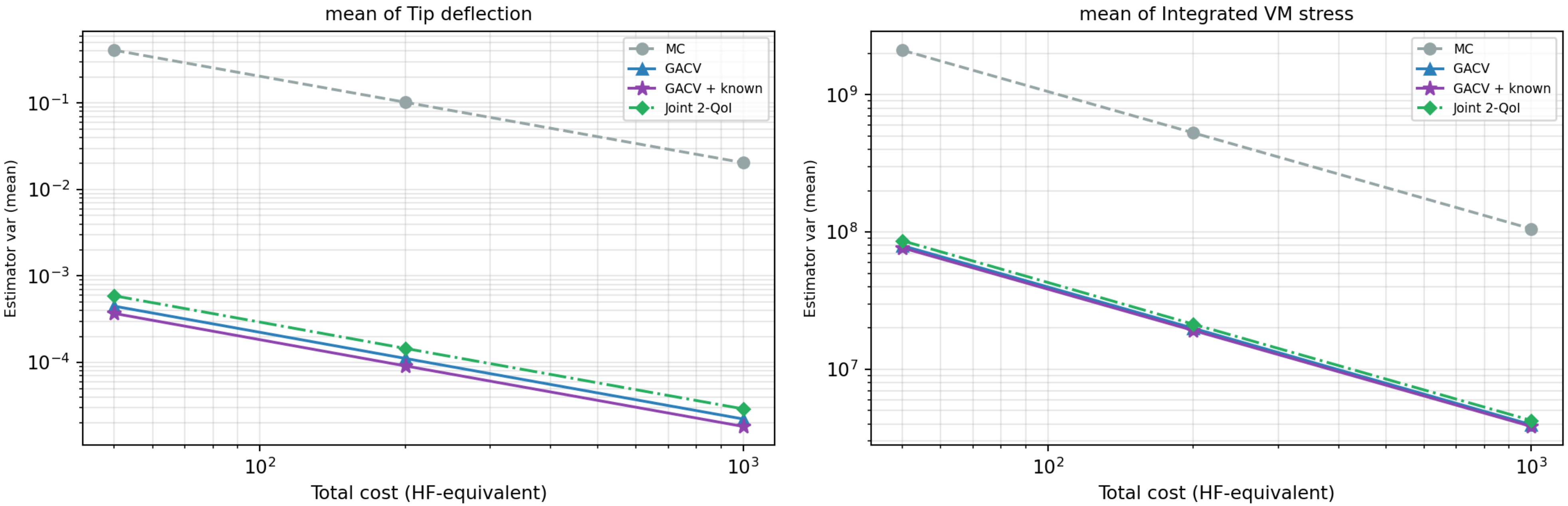
Pilot correlation over every (model, QoI) pair: deflection correlations sit near one for the physics models and 0.95 for the surrogate, while stress is harder (0.97 and 0.85).

The Headline: Variance vs Cost

At every budget, GroupACV reaches a far lower estimator variance than Monte Carlo.

Known mean helps. The surrogate's analytic mean lowers the variance further — most for tip deflection.

Two to three orders. GroupACV sits 2–3 orders of magnitude below MC for the mean of each QoI — unbiased by construction.



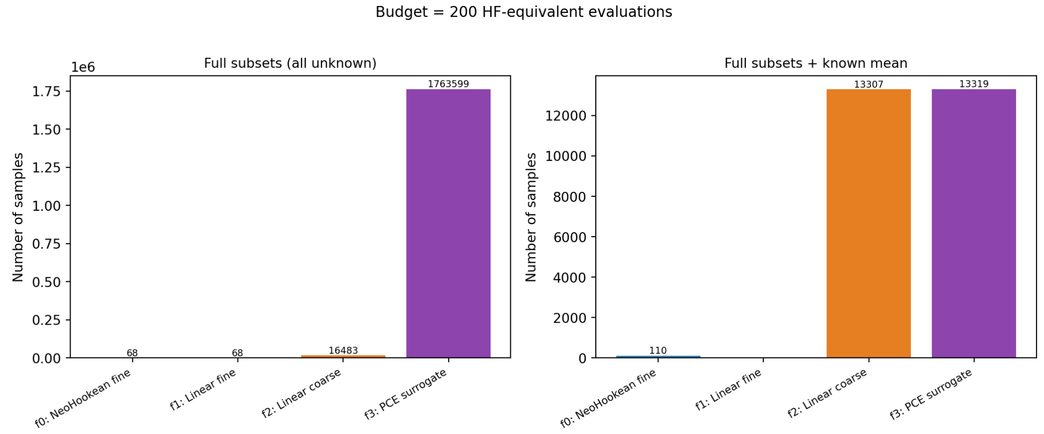
Estimator variance vs total cost for the mean of each QoI: GroupACV (blue) sits 2–3 orders below Monte Carlo (dashed); the known surrogate mean (purple) lowers it further, most for tip deflection.

Where the Budget Goes

The optimizer spends where evaluations are cheap and informative.

All unknown. The near-free surrogate absorbs ~1.8M samples and the coarse model ~16k, while the HF reference takes only ~68 — just enough to anchor unbiasedness.

Known mean. Knowing the surrogate's mean frees those samples: its count collapses to ~13k and the HF roughly doubles (to ~110), tightening the control-variate correction at the same budget.



Optimized allocation at a fixed budget, without (left) and with (right) the surrogate's known mean. Note the y-axes differ: surrogate sampling falls from ~1.76M to ~13k, freeing budget that lifts the HF count from 68 to 110.

Open Frontier: Explore vs. Exploit

Choosing the pilot size optimally needs the very covariance the pilot is meant to estimate — so balancing exploration against exploitation is still open.

Models as arms. Each model is a slot machine: a coin (its cost) buys a pull; the payout is the error it removes. Which arms to keep pulling?

AETC. Adaptive Explore-Then-Commit casts this as a multi-armed bandit — today for single-mean estimation only.

Automating the split. Dixon, Gorodetsky, Jakeman, Narayan and Xu (2025) balance explore and exploit iteratively; general multi-statistic and multi-output automation remains open.

Each model is an arm: the coin is its cost, the screen its mesh fidelity, the cheque the error it buys down.

Recap: When to Reach for This

One framework has carried the whole tutorial — every estimator is a choice of groups.

Default to all-subsets GroupACV.
It contains MLMC, MFMC, ACVMF and MLBLUE as special cases — its optimized variance is a lower bound on all of them, and the search costs only optimiser time.

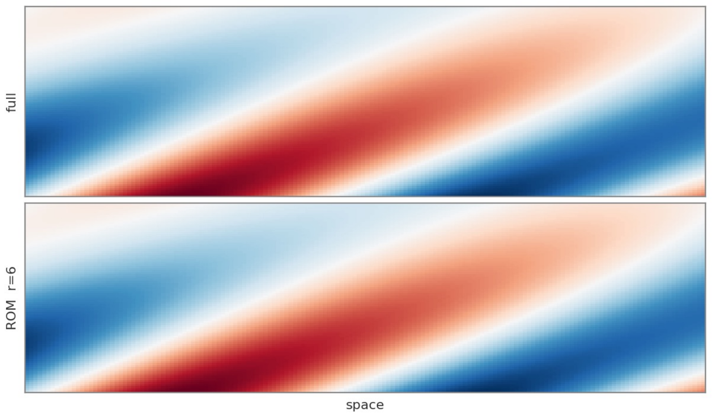
Match the structure to your models.
Clean cost hierarchies suit MLMC and MFMC; partial orders — a refinement pair here, a load-dependent pair there — need GroupACV or MLBLUE.

Exploit what you already know.
A surrogate with an analytic moment enters as a mixed-known model and strictly lowers variance — a surrogate is just another model in the ensemble.

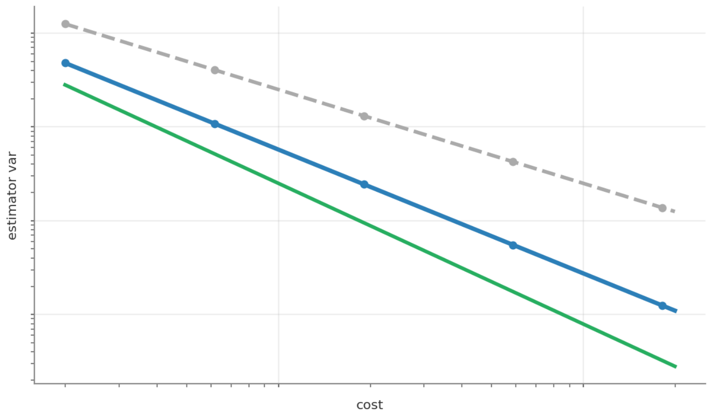
The variance wall, control variates, approximate control variates, group structures, and the extensions — one ladder, climbed.

One Toolkit, Many UQ Workflows

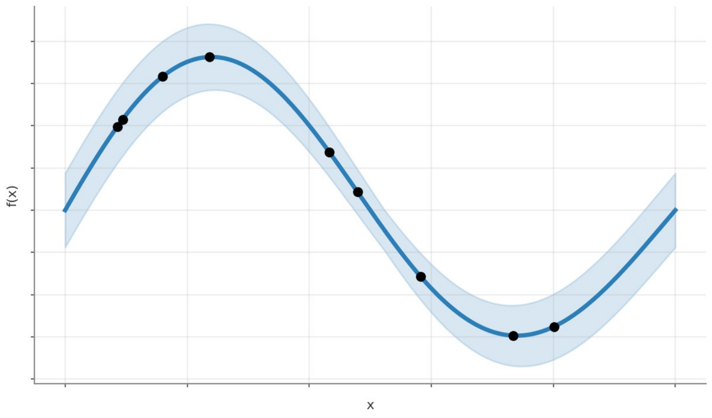
Multi-fidelity estimation is one capability in an open-source Sandia toolkit — 100+ executable, dual-backend tutorials spanning the UQ workflow.
sandialabs.github.io/pyapprox



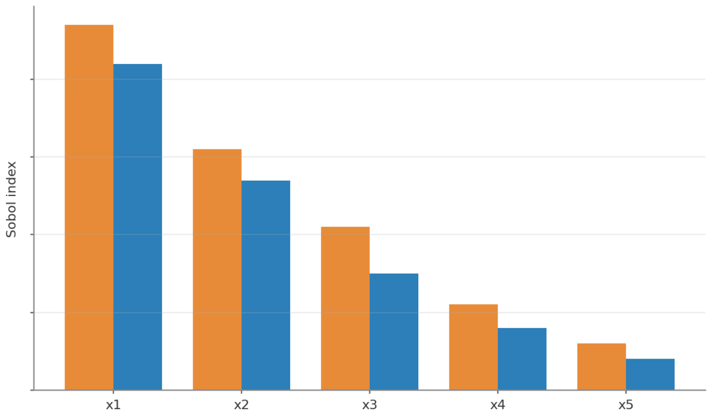
Operator / Data-Only ROM



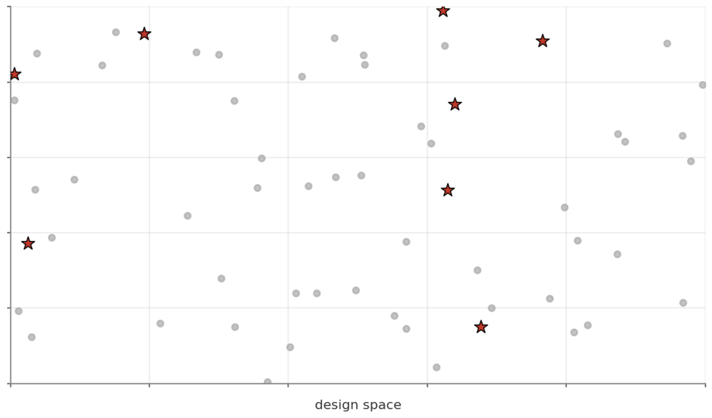
Multi-Fidelity Estimation



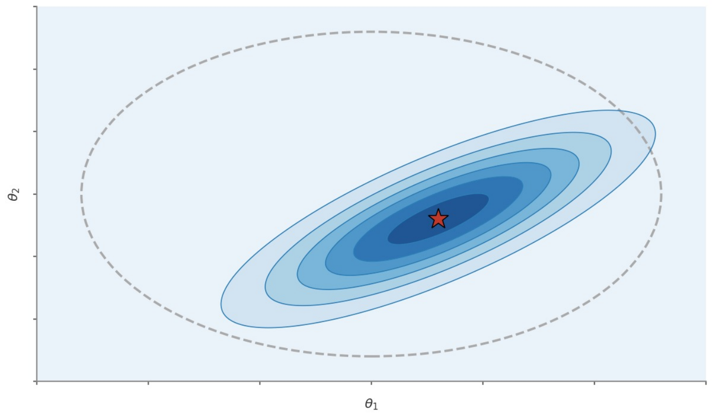
Surrogate Modeling



Sensitivity Analysis



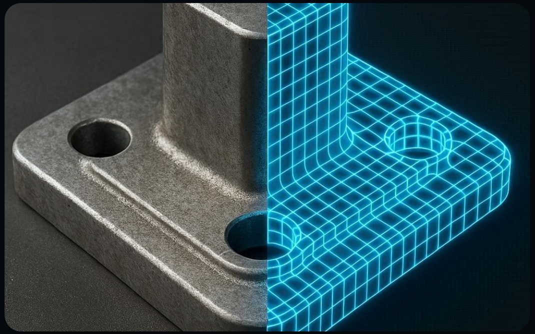
Optimal Experimental Design



Inverse UQ

Where Simulation Meets Reality

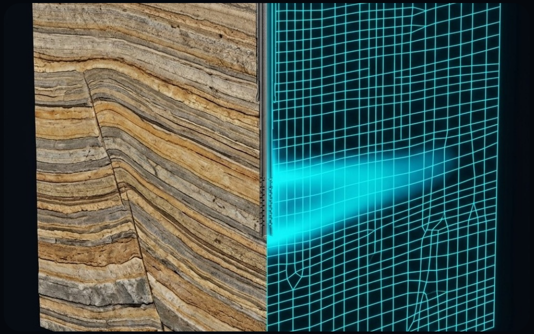
PyApprox is used across domains with expensive simulations, uncertain parameters, and high-stakes decisions.



Structures & Fatigue

Stress, deformation & service-life prediction

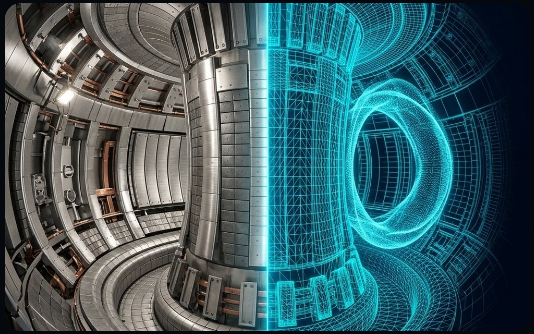
Elasticity / plasticity



Subsurface & Resource Extraction

Oil, gas & geothermal recovery

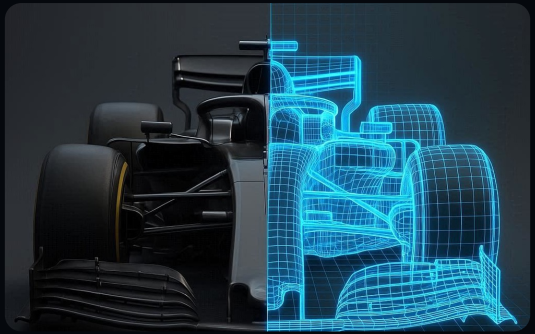
Darcy flow in porous media



Nuclear & Fusion

Reactor margins & magnetic-confinement design

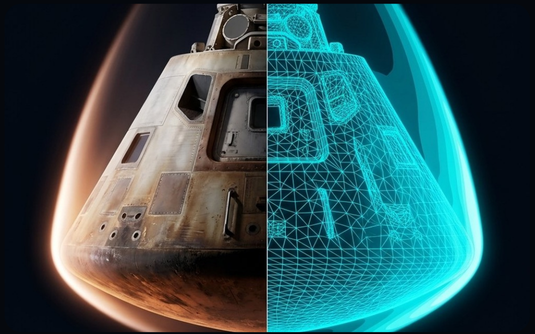
Neutron transport / magnetohydrodynamics



Aerodynamics

External flow & vehicle performance

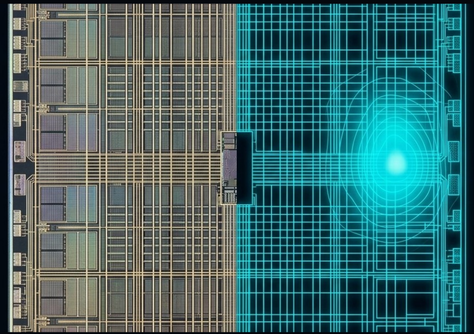
Navier–Stokes



Hypersonics & Reentry

Aerothermal heating & shock prediction

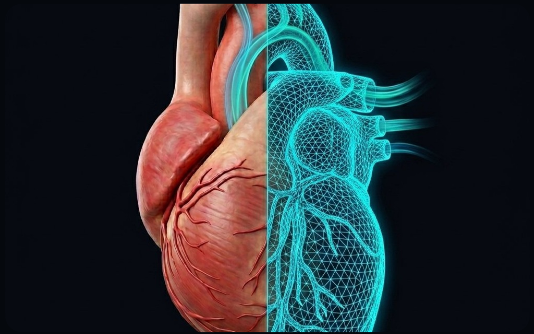
Reacting compressible flow



Semiconductors & Electronics

Electro-thermal design & reliability

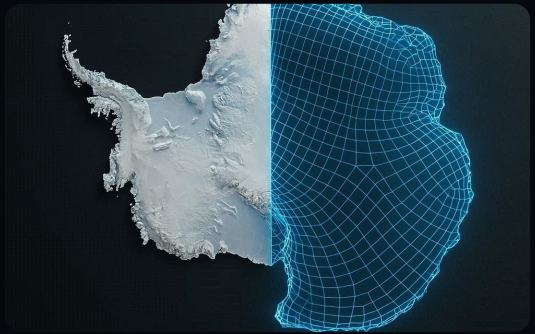
Drift–diffusion + heat



Biomedical

Hemodynamics & patient-specific modeling

Navier–Stokes + fluid-structure interaction



Earth Systems

Ice-sheet projections and national security impact assessment

Nonlinear Stokes flow

Want More Details?

PyApprox

Faster, Defensible Delivery via Simulation & Data

Deep dives into today's topics — and well beyond. Every tutorial is executable, dual-backend, and reproducible.

sandialabs.github.io/pyapprox

github.com/sandialabs/pyapprox

Scan for docs & tutorials

References

Multi-Fidelity Estimators

Giles (2008). Multilevel Monte Carlo path simulation. *Operations Research 56(3):607–617.*

Peherstorfer, Willcox & Gunzburger (2016). Optimal model management for multifidelity Monte Carlo estimation. *SIAM J. Sci. Comput. 38(5):A3163–A3194.*

Gorodetsky, Geraci, Eldred & Jakeman (2020). A generalized approximate control variate framework for multifidelity UQ. *J. Comput. Phys. 408:109257.*

Schaden & Ullmann (2020). On multilevel best linear unbiased estimators. *SIAM/ASA J. Uncertainty Quantification 8(2):601–635.*

Gorodetsky, Jakeman & Eldred (2024). Grouped approximate control variate estimators. *arXiv:2402.14736.*

Dixon, Gorodetsky, Jakeman, Narayan & Xu (2025). Balancing exploration and exploitation to automate multi-fidelity estimation. *arXiv:2505.09828.*

Multi-Fidelity Surrogates

Kennedy & O'Hagan (2000). Predicting the output from a complex computer code when fast approximations are available. *Biometrika 87(1):1–13.*

Perdikaris et al. (2017). Nonlinear information fusion algorithms for data-efficient multi-fidelity modeling. *Proc. R. Soc. A 473:20160751.*

Cutajar, Pullin, Damianou, Lawrence & González (2019). Deep Gaussian Processes for Multi-fidelity Modeling. *arXiv:1903.07320.*

Software

Jakeman (2023). PyApprox: sensitivity analysis, Bayesian inference, optimal experimental design, and multi-fidelity UQ and surrogate modeling. *Environmental Modeling & Software 170:105825.*

sandialabs.github.io/pyapprox

MIT-licensed · NumPy & PyTorch · 100+ tutorials