# SEARCHING HARDER HURTS AT A FIXED BUDGET: THE UNREPORTED CANDIDATE-SET SIZE IN ADAPTIVE COLLOCATION

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

## ABSTRACT

Adaptive collocation for physics-informed neural networks picks $B$ training points by scoring $V$ candidates and keeping the best. Across this literature $V/B$ spans four orders of magnitude and is almost never reported. It is not only a cost knob: keeping the best $B$ of $V$ is truncation selection at threshold $B/V$, and the optimism of a maximum grows with the number of items scored, so $V$ sets how hard the criterion is optimised. We sweep $V/B$ at a fixed budget of $40$ points over $64$ shared seeds, against a uniform control re-drawn on the rules' own schedule. Two rules with unrelated scores, a residual magnitude and a training-dynamics score, are both level with that control at $V/B = 1.25$ and both fail every seed by $20$, on a sharp one-dimensional fit and on a Poisson problem, where that control holds at $1$ in $64$: what collapses is the selection form, not the criterion. Published rules that soften that same maximum fail later and less, under $24$, $19$ and $8$ of thirty analysis choices on the first of those. Let the set grow instead, as the shipped implementation does, and the same rule takes no harm from $V$; held at a fixed budget of its own final size it fails all $32$ seeds of a larger pool again. The harm is searching hard at a fixed budget, not searching hard. Which rules are exposed starts with one question: does selecting a point remove its own score? Necessary, not sufficient: our own arms subtract and collapse anyway; the objective has to be a function of the selected set. The effect survives four times the budget and seventeen times the pool, where the rule whose objective is defined on the selected set stops being immune; a registered attempt at $2{,}000$ points missed its own accuracy gate. Report $V/B$, and re-draw the control.

## 1 INTRODUCTION

Here is the reference implementation of residual-based adaptive refinement, one standard way to choose collocation points for a physics-informed neural network (Raissi et al., 2019), from the library maintained by an author of the method (Lu et al., 2021):

```
X = geomtime.random_points(100000)   (candidates)
   data.add_anchors(X[x_id])   (one point added)
```

One hundred thousand candidates are scored so that a single point can be added. It is not an implementation detail. Hold the point budget $B$ fixed and let $V$ be the number of candidates the rule is shown. Keeping the best $B$ of $V$ places the selection at the $1 - B/V$ quantile, so $V$ is not a cost knob but *how hard the criterion is optimised*, and $V/B$, candidates inspected per point kept, is the unit that compares one paper's setting with another's.

And almost nobody reports it. The rule is Lu et al. (2021)'s, whose own statement adds the $m$ largest residuals in "a set of randomly sampled locations $\mathcal{S}$" and leaves $|\mathcal{S}|$ unspecified; Wu et al. (2023) restate it as "a set of dense points" and Florido et al. (2024), three years later, as "a fine grid of random points", neither with a size. PINNACLE fixes its candidate pool at about $4\times$ the selected budget (Lau et al., 2024), in an appendix, with a sentence of reason and, for its sampling variant, one sentence naming robustness to pool size as future work. Across published methods the ratio of candidates to points kept spans four orders of magnitude, and none measures what changing it does.



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.

The closest attempt sweeps the resolution of a sampling density rather than the number of candidates competing, and concludes the opposite (Nabian et al., 2021; Appendix P).

The identity is exact for `add_anchors` above, which takes a literal best $m$. The published softened arms below reach it through the score's distribution rather than an $\arg\max$, which weakens the identity; our own two take an $\arg\max$ at every pick (Appendix P).

**Optimising the criterion harder makes the trained network worse.** On `crenel`, a six-block square wave with five interior jumps, at $B = 40$ and 64 seeds shared across every cell, PINNACLE fails on 0 of 64 seeds at $V/B = 1.25$ and on 30 at $V/B = 30$, against the *held* uniform control at 2 throughout. It crosses from better than that control to worse at $V/B = 5$ (paired Wilcoxon $p = 1.7 \times 10^{-4}$, Holm-corrected over that panel's thirty tests), and BADGE crosses at the same ratio. Nothing about the rule changed.

**And they are not failing to optimise.** Under gradient flow restricted to $S$, $e^\top K[:, S]\, e_S$ is exactly the instantaneous decrease of the full-domain error: derived, and no deployed rule's implemented score, though dynamicAL's sums its two pieces. Once the residual has localised, a larger candidate set buys PINNACLE five times as much of it on `poisson1d` while the trained network is worse. The divergence is not between a rule and its target, but between the one-step objective and the trained outcome: the shape familiar from reward-model overoptimisation (Gao et al., 2023) and the beam-search curse (Koehn & Knowles, 2017; Stahlberg & Byrne, 2019), read closely in Appendix W, with the two papers that make that curse go away.

**Which rules suffer is predictable from their definitions, and the prediction is derived rather than fitted.** The error a run ends at is a set function of $S$ through $K_S^{-1}$. Call a criterion **pointwise** when it scores each candidate on its own and keeps the best (RAD, PINNACLE and BADGE as deployed, and RAR-G and dynamicAL exactly), and **set-level** when its objective is defined on the selected set and optimised over sets (GRAD-MATCH alone here). A pointwise criterion is a proxy for that set function, with a quantile that $V$ moves. One test is necessary, not sufficient: does *selecting a point remove the score at that point*? Its formal name is adaptive submodularity (Golovin & Krause, 2011); what is new here is reading it off a deployed rule before a run rather than proving it of an objective. The residual there decays by $1 - e^{-\lambda t}$ over the horizon, so a score built on mass at $\lambda t \lesssim 1$ is not removed, the $\arg\max$ does not move, and a larger candidate set buys near-duplicates of it.

In kernel interpolation the removal is exact, and there a pointwise score is not merely safe but optimal: $P$-greedy's points are asymptotically uniform for Sobolev-space kernels (Santin & Haasdonk, 2017). And $f$-greedy, which is RAD's criterion, is $L^p$-optimal there up to a poly-logarithmic factor for $p \in [1, 2]$ and faster for $p > 2$ (Wenzel et al., 2023) (Appendix W.1). A set-level objective removes its own score *within* a round, which is why scoring the selected set is a readable proxy for the test and why GRAD-MATCH survives every sweep at $N = 1{,}200$; it is only a proxy, and it breaks by $B = 80$.

Figure 1 is the whole paper: where the kept points go as $V$ grows, one rule collapsing in $V$ at a fixed budget, and the same rule taking almost no harm when the set may grow.

**Why this was not measured before.** $V$ went unmeasured for three reasons visible in the papers themselves: it has no name, being the length of an intermediate array rather than a row in a hyperparameter table; it is conflated with the discretisation $N$, which equals $V$ only for rules that rank the whole pool; and seeing it needs an axis this literature does not use, since at the 3 to 10 seeds standard here 0 to 30 failures in 64 hides inside a mean.

**Contributions.**

1. **At a fixed budget of 40 points, raising $V/B$ from 1.25 to 30 takes this literature's reference implementation from 1 failure in 64 seeds to 64 on `poisson1d`, while its re-drawn uniform control stays at 1.** That is `add_anchors` of the largest residuals, climbing through 23 and 58 (Figure 1); on `crenel` it goes $0/64$ to $64/64$ against a control at 0. **And it outlives the budget.** At $B = 160$ and $N = 20{,}000$, four times the points and seventeen times the pool, uniform fails 0 of 32 at every rung while raising $V/B$ from 2.5 to 30 makes PINNACLE fail 5 of 32 and GRAD-MATCH 7, a within-rule change of $-1.42$ and $-2.45$ at $p \leq 10^{-4}$ and RAD

$-0.70$ without significance. *Searching harder breaks a problem that random sampling solves.* The larger is the set-level rule, the cell Contribution 2 bounds itself with (Appendix X).

2. **A dividing line derived rather than fitted to the rules it sorts: the five pointwise rules this literature supplies all end the two ladder tasks worse than they start them, and the one set-level rule ends better.** The terminal error of the linearised model is a set function, and its first-order truncation in the horizon is a sum of per-point terms whose diagonal part is the score this literature deploys, whose modularity is machine-checked in Lean 4 with no `sorry` and no added axiom, so a rule's exposure is readable from the rule *as implemented*, including for rules we did not run, with two exceptions named in Appendix AA. Both sides of the line fail on a saturated task. Our own two arms have that property and cross anyway, so it is not what protects the set-level rule (Appendix K). $B = 40$ **is a condition of the sorting, and here is its limit**: at $B = 80$ and 160 on `poisson1d` the set-level rule loses its immunity on the failure count, at $B = 400$ the ordering inverts outright, and on three of the five ladder tasks the effect is in the third decimal. We have a repair for the inversion and no mechanism (Appendices X, K and M).

3. **Which uniform control you use decides which rules cross, and the right one is already published: held and re-drawn uniform fail** $35/64$ **and** $1/64$ **on `poisson1d`.** The re-drawn one is Wu et al. (2023)'s Random-R and Daw et al. (2023)'s dynamic baseline, in papers we benchmark, at the same order-of-magnitude gap they report. Ours is only that *swapping* it changes which rules cross a $V$ ladder (Appendix O). **The same budget spent on coverage repairs a rule, not just a control**: one line on RAD, half its points reserved for uniform coverage, takes 28 failures in 500 runs to 9 at the same 1.2 GFLOP, and no sampled uniform budget matches it below $1.54\times$ its compute under either control or statistic. **Its price is accuracy**: GRAD-MATCH leads it on the paired mean on all five tasks at $24\times$ the selection cost (Appendix N).

4. **The budget is the third condition: the rule that fails all** 64 **seeds at a fixed** $B = 40$ **fails at most** 2 **of** 64 **at any rung when it *appends* instead, as the shipped library does.** It beats appending uniform points by $+0.05$ to $+0.40$. Held at a fixed budget of the growing arm's own final size it fails 32 of 32 again, at the larger pool of Appendix L, so it is the growth and not the budget. What this paper indicts is fixed-budget selection, not the shipped procedure (Appendix L).

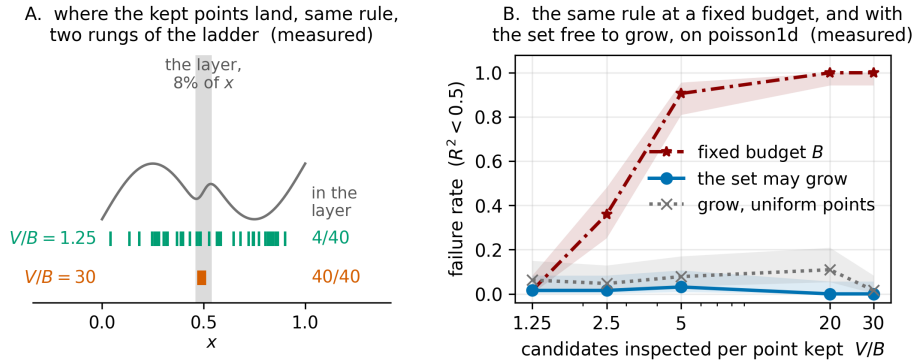



Figure 1: **The paper in one figure. (A)** One window of $V$ drawn from the pool of 1,200 and the largest $B = 40$ residuals kept. At $V/B = 1.25$ four of the forty land in the layer, 8% of the domain; at $V/B = 30$ all forty do. Over 64 draws the means are 3.7 and 40.0, with 9.7 and 2.0 of ten equal-width bins occupied (`scripts/fig_where.py`). **(B)** One rule, `add_anchors` of the largest residuals, on `poisson1d` over 64 shared seeds, run two ways: at a fixed $B = 40$ it fails 1 of 64 at $V/B = 1.25$ and 64 of 64 by 20, and letting the set grow from 40 points to 80, as the library shipping it does, it stays between 0 and 2 and under appending uniform points at every rung. Bands are 95% Wilson; `crenel` gives a cliff and two flat zeros (Appendix L; `scripts/fig_spine.py` on `log{ET,EV}_poisson1d_V{V}`).

## 2 SETUP

**The selection loop.** An *arm* is one run configuration: a rule, a control, or a rule at another budget. A network $f_\theta$ is trained on a subset $S \subset \{1, \dots, N\}$ of a candidate pool, $|S| = B \ll N$, and the subset may be revised every $c$ optimiser steps. Every comparison fixes $B$, $c$, the architecture and the number of steps, so two arms differ only in which points are chosen and what the choosing costs, with three exceptions: the control draws once and holds it where every rule re-selects (Appendix O), in Table 1 our own arms score a $V = 200$ window where the other rules score all $N$, an asymmetry Section 4's ladder removes (Appendix P), and our own arms carry half the budget between rounds where the published ones rebuild it (Appendix K).

Under gradient flow the pool residual obeys $\dot{e} = -Ke$, where $K = JJ^\top$ is the empirical neural tangent kernel (Jacot et al., 2018). Point $j$'s share of the loss decrease is therefore $s_j = e_j(Ke)_j$: its own error, times the error reaching it through the kernel. Residual rules use $e_j$ alone and diagonal rules $e_j^2 K_{jj}$, so both drop the off-diagonal (Appendix B). And $K$ drifts throughout training, provably so at infinite width once the operator is nonlinear (Bonfanti et al., 2024), which is why re-selection is part of the protocol rather than a detail.

**Tasks.** Eleven problems, typed plainly because five are not PDEs. Six are PINN problems: `poisson1d` (a static sharp interior layer), `nlrd1d` (a nonlinear reaction–diffusion front), and four space–time $(x, t)$ problems, `vburgers1d`, `burgers1d`, `allencahn1d` and `convdiff1d`. Four are one-dimensional regression benchmarks with sharp structure (`sinjump`; `crenel`, a six-block square wave; two more in Appendix K), and one is binary classification. Three more carry numbers here (Appendix A). Task counts differ by claim; Appendix A lists the tasks and seeds behind each, and what this count leaves out.

**The regime, measured rather than asserted.** Selection can only matter when the budget is small relative to the solution's structure, so we locate that boundary: $B$ from 20 to 320 at 32 seeds. **Selection stops changing whether a run fails by $B \approx 80$.** At $B = 20$ a held uniform draw fails $10/32$ on `crenel` and $25/32$ on `poisson1d`, a re-drawn one $0/32$ and $9/32$ (Appendix N); at $B = 80$ it fails $0/32$ and $2/32$, and from $B = 160$ $0/32$ on both. Past $B \approx 80$ it is the *control* that stops failing: arms still separate from it on the paired statistic, and at $B = 160$ on `poisson1d` on the count too (Appendix X). It does not stop buying accuracy: the paired gap to GRAD-MATCH decays but stays positive to $B = 320$ (Appendix X), so our regime claim rests on the failure rate, and past $B \approx 80$ a practitioner trades a small paired gain against selection cost. Our failure-rate claims are all below it; above it we read paired gaps, to $B = 400$.

**A second boundary is the optimiser, and testing it moved it.** We called low-viscosity Burgers and Allen–Cahn *optimisation*-limited under Adam and `convdiff1d` saturated, and rested no claim on the three. Their sharp fronts are what the second condition is about, so that was a live objection; we ran the first and it was wrong. `burgers1d` at $\nu = 0.01/\pi$ is selection-limited, carries a headline result and harms our own arms too (Appendix M); `allencahn1d` we also ran, and it stays out for the other reason, saturation, with a control that fails $0/64$ at every rung (Appendix AB).

**Scale, stated once and not softened.** Two hidden layers of 64 units, 4000 Adam steps, $B = 40$, $c = 100$, pools of 1,200 and 20,000, well below the field's benchmark scale, which Section 7 prices. Every arm sits at the same scale, so the comparisons are internally valid where the numbers are not. Cost is charged by an analytic FLOP model and by wall clock, which disagree by two to four times, against us (Appendix H).

**Reporting.** Where selection matters a run's outcome is close to two-valued, so a mean over three to ten seeds, which is what this literature reports (Wu et al., 2023; Daw et al., 2023; Hao et al., 2024; Krishnapriyan et al., 2021), is a failure rate with variance bolted on. We import the optimisation literature's protocol instead (Rohrhofer et al., 2023). **We use 100 seeds per cell in Table 1** and 64 or 32 in the $V$ and budget sweeps, with the mechanism sweeps, the classification control and the ablation at 8, 16 and 5, and the concentration mediator at 6 seeds by 12 draws, listed per claim in Appendix A. A cell is **void** when the control fails $0/n$ or $n/n$, nothing then being winnable against it; within-rule comparisons across $V$ are exempt. Intervals written $\pm$ are 1.96 SE on a paired mean, bracketed ones on proportions are Wilson, and counts use a two-sided Fisher exact test, which discards the pairing.

Paired tests are a tie-corrected normal approximation (Appendix Q). We lead with $\Pr[R^2 < 0.5]$, a threshold conservative against the field's, with means and medians beside it (Appendices Q, S). The protocol is $B = 40$, $N = 1{,}200$, 4000 steps and re-selection every 100, with three departures: the smoothed-bump replication at $B = 10$, the capacity ladder, and the budget sweep at $B \in \{80, 160\}$, $N = 20{,}000$.

## 3 THE HEAD-TO-HEAD AT 100 SEEDS

**Two arms we modified are much worse than no selection.** On `crenel`, the held draw fails on 3 of 100 seeds against our PINNACLE ablation's 46 and BADGE's 36 at a fixed budget its published form does not use, while the remaining arms improve on it, bar RAD's $9/100$ ($p = 0.13$). Being beaten by the control happens on `crenel` against the held draw; on `poisson1d` it happens against the re-drawn control by failure rate, though not by the paired test (§7).

**Our own coverage floor on RAD is never worse on the failure count than the published rule it modifies, and better on three of five, significantly on two**: RAD's 28 failures in these 500 runs become 9 at the same 1.2 GFLOP of selection, $9/100 \to 1/100$ on `crenel` and $6/100$ on `poisson1d` from $16/100$, against our arms' $15/100$ and $12/100$, at 1.2 GFLOP against 5.2. The arm that beats it, at 1, selects with $24\times$ the compute.

Table 1: Failure rate at 100 seeds per cell, lower better: seeds below $R^2 = 0.5$, a relative $L^2$ error of 0.50–0.71. $B = 40$ **kept from** $N = 1{,}200$**,** $[64, 64]$ **with** tanh**,** 4000 **Adam steps, re-selection every** 100**; our two score a** $V = 200$ **window and the other five all** $N$. `random` is the *held* draw and `re-drawn` the one Contribution 3 argues for, at 64 seeds on the two tasks Section 4's ladder runs (Appendix O). **Not an equal-**$V$ **comparison**; Appendix R prints the three tasks at one $V$ and loses us the `poisson1d` column. "PINNACLE, ours abl." is *our* ablation of that method, its pool subsampling removed, and is not a measurement of the published rule; `RAD+cov` is our one-line coverage floor on RAD (Appendix P). Selection GFLOP, also lower better, is the run's total minus the training term (Appendix H). Paired differences: Appendix R. Generated by `scripts/main_table.py` from `runs/logDB_{task}_s100`.

| method | crenel | nlrd1d | poisson1d | sinjump | vburgers1d | sel. GFLOP |
|---|---|---|---|---|---|---|
| random | 3/100 | 9/100 | 52/100 | 1/100 | 3/100 | 0.0 |
| re-drawn | 0/64 | – | 1/64 | – | – | 0.0 |
| RAD | 9/100 | 2/100 | 16/100 | 0/100 | 1/100 | 1.2 |
| RAD+cov (ours) | 1/100 | 2/100 | 6/100 | 0/100 | 0/100 | 1.2 |
| PINNACLE, ours abl. | 46/100 | 3/100 | 14/100 | 0/100 | 0/100 | 29.6 |
| BADGE | 36/100 | 4/100 | 20/100 | 0/100 | 5/100 | 28.9 |
| GRAD-MATCH | 0/100 | 0/100 | 1/100 | 0/100 | 0/100 | 28.9 |
| ours (addpiv) | 1/100 | 0/100 | 15/100 | 0/100 | 0/100 | 5.2 |
| ours (kepiv) | 0/100 | 0/100 | 12/100 | 0/100 | 0/100 | 5.2 |

**What the fix costs.** A corollary, not a result; Mirzasoleiman et al. (2015) own the efficiency claim. At $B = 40$ PINNACLE, GRAD-MATCH and BADGE each spend 88% of a run's operations, and windowing them to $V = 200$ stops that share growing with the pool. **Cheap is not safe**, since that window is $V/B = 5$, their crossing rung; the 88% prices our budget, not the rules, being 3 to 4% at Hao et al. (2024)'s budgets; and at equal compute uniform selection with more points beats four of six arms, three if the control is held (Appendices F, N).

## 4 THE CURVE: ACCURACY AGAINST CANDIDATES PER POINT

Figure 2 is this section, measured on the published rules and not on a method of ours. At a fixed budget, pointwise criteria degrade as the candidate set grows while a set-level one does not: over six tasks the set-level rule improves on five and is flat on the sixth, and of the eighteen pointwise cells thirteen degrade, three are null and two improve, both on `staircase`. On three of them, one the canonical Burgers instance, the degradation crosses below the re-drawn uniform control. Every task here runs at $B = 40$, one to two orders below the collocation budgets those papers use, the effect holding to $B = 160$ (Appendix M).

**The intervention is one method's existing design choice, applied to the others.** PINNACLE already draws a candidate pool "around $4\times$ as large as the selected points budget in each round" and names robustness to it as future work (Lau et al., 2024), so restricting the candidate set is not our idea; measuring what it does is. **Redrawing also churns the set**, and holding one window all run rules that out, five of six cells null and the sixth $-0.004$ (Appendix Y). The sweep is $V \in \{50, \ldots, 1200\}$ at $B = 40$ and $N = 1{,}200$, so $V/B$ from 1.25 to 30, on the two tasks with room for a rate test. Ours replace $B/2$ a round, so their realised ratio is twice the plotted one (Appendix D).

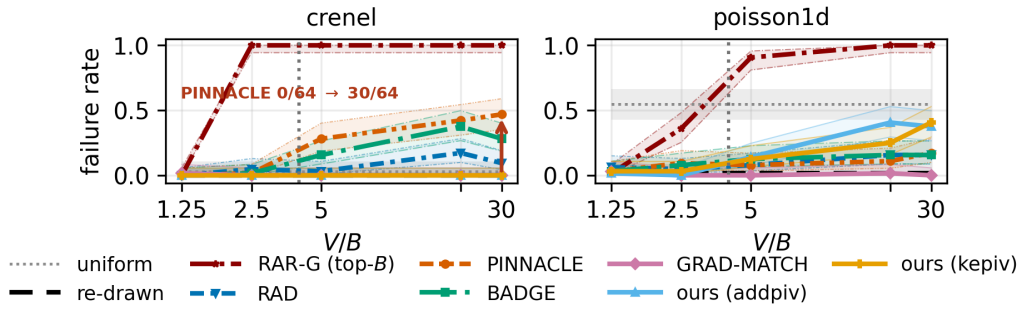



Figure 2: **Pointwise rules get worse as the candidate set grows; the set-level rule does not.** Failure rate against candidates inspected per point kept, at a fixed budget of $B = 40$ with 64 seeds shared across every cell; bands are 95% Wilson intervals. **Read the dash-dot curves against the dashed black line**: the pointwise rules against the control re-drawn on their own schedule, which Appendix O argues is the comparison this axis is entitled to. Every dash-dot curve ends higher than it starts and all four pointwise rules cross that control, $0/64$ on `crenel` and so one-sided. Solid marks arms that cross on neither panel: on `poisson1d` both of ours rise, $1/64$ to $24/64$ and $2/64$ to $26/64$, and GRAD-MATCH alone is flat on both. The faint dotted line is the *held* draw; §3 gives both rates. Neither control reads $V$. The vertical dotted line is PINNACLE's own default, $V/B = 4$.

**The sign flips on both panels, and the rule it flips hardest for is the one in the snippet of Section 1.** On `crenel` the candidate set decides the *sign* for the pointwise rules: PINNACLE fails on 0 of 64 at $V/B = 1.25$ and 30 at 30, against the *held* control at 2 throughout. PINNACLE and BADGE cross below it at $V/B = 5$ and RAD at 20 and not at 30; the first two are one crossing and not two, their scores being one operator (Appendix P). The reversal needs no threshold: the paired mean of (rule $-$ uniform) in $R^2$ ends negative for all three, each starting significantly *above* the control, against $+0.187$ for GRAD-MATCH (Appendix E). The $p$ is the paired Wilcoxon Holm-corrected over that panel's thirty tests, quoted *at each rule's crossing rung*. One rung further out RAD's reversal reaches only 0.059 and does not survive (Appendix E). Against a control *re-drawn* at every re-selection the rungs hold, the far end falls further, and PINNACLE's start loses significance (Appendix O); at $0/64$ it can show harm and not help here, so the counts above use the held draw. The same ladder on `burgers1d` at $\nu = 0.01/\pi$, the instance this literature benchmarks on, is the second crossing task: six of seven arms degrade in $V$, five cross below the re-drawn control uncorrected and two survive Holm, and the one that does neither is GRAD-MATCH (Appendix M).

**On `poisson1d` the difference was the control.** Its held draw fails 35 of 64, a bar a rule clears cheaply. Against a control re-drawn on the rules' own schedule, at 1 of 64, seven of the eight rule arms cross on an uncorrected Fisher test and *four* survive Holm: the two top-$B$ rules below and both of ours. That panel adds dynamicAL, the top-$B$ rule Figure 2 does not plot. **Every panel's multiplicity family is that panel's own arms by its own rungs**, and every corrected $p$ we quote is Holm over it, so the size follows the panel: thirty for the paired means over six arms, forty-five and forty for the two count panels' nine arms and eight. Each appendix states its own. No published softened rule crosses there under correction, so what changes on this panel is which rules are exposed, not whether any is. The within-rule ratio of best rung to worst, 3.3 to 5.5 on that task, takes no control.

**And none of the four softened arms is a top-$B$ rule**, which is what the identity is exact for. RAD samples from a density, PINNACLE and BADGE take one $\arg\max$ then $k$-means++, GRAD-MATCH is matching pursuit. Two arms are: RAR-G, `add_anchors` of the $m$ largest residuals (Lu

et al., 2021) and the rule in the opening snippet, and dynamicAL (Wang et al., 2022a), whose score is not a residual magnitude. **Both are level with the re-drawn control at** $V/B = 1.25$ and fail every seed on `crenel` from 2.5 and on both from 20, crossing at 2.5 on each, earlier than any softened rule. Two unrelated scores collapsing at one rung is the selection form, not the field it scores.

**And the collapse is the fixed budget, not the rule.** RAR-G as Lu et al. (2021) ship it *appends* its point to a set that keeps everything it had, and run that way it does not collapse at any rung of either task (Appendix L). **It is the growth and not the budget**, registered before it ran: held at a fixed $B = 80$, the growing arm's own final size, the same rule fails 32 of 32 by $V/B = 30$ on both tasks (Appendix L).

**GRAD-MATCH is the exception in both panels here, and the exception has a limit.** Alone among the rules here it optimises a set-level objective over sets, a full-data gradient match rather than diversity over a pointwise score, so no per-point score survives the round. Changing the other points' scores is not what buys it: our own arms do that and still collapse on `poisson1d`, where GRAD-MATCH is the only arm that improves (Appendix K). That is a property of the round, not the pool, and past its limit the order does not weaken, it inverts. On `poisson1d` GRAD-MATCH stops being the protected one on the failure count, at both $B = 80$ and $B = 160$: it fails $8/32$ and $7/32$ against PINNACLE's $5/32$ at both, and on the within-rule change in test $R^2$ across $V$ at 160 it is $-2.45$ against PINNACLE's $-1.42$ though at 80 PINNACLE's is the larger, $-1.65$. On `crenel` the inversion is a tie, GRAD-MATCH and RAD joint-worst at $-0.009$. We have a repair but no mechanism: diluting the selection with uniform draws removes the damage and a registered ablation refutes our account of why (Appendix X), and it is the reason the sorting in Contribution 2 is a $B = 40$ statement.

**What we could not detect is an effect of the pool at fixed $V$.** Raising the pool from $N = 1{,}200$ to $20{,}000$ changes nothing, and the two groups say different things. For the whole-pool arms $V = N$, so the same cell scores 16.7 times as many and moves nothing: $28/128$ against $26/128$, odds ratio $0.911$, $[0.476, 1.736]$, by a rule fixed in advance and the only cell here with power. The windowed arms hold $V$ while $N$ rises and must not move, and they do not: the two windowed rules fail $0/64$ at both pool sizes, and the one failure in that group at each is the control's own. A null with no power of its own. That is not saturation but one null whose interval permits the odds to halve or rise by three quarters, and it is all we have above $V/B = 30$ (Appendix T). One registered prediction of ours failed here, that the published rules would be flat in $V$ down to $V \geq 100$: on `crenel` they are not.

## 5 Where the damage is, and what mediates it

**Both conditions, that selecting a point leaves its own score and that the error is irreducible, come from one expression, which is not ours.** Under gradient flow restricted to $S$ the pool residual moves as $\dot{e} = -K[:, S]\, e_S$, so $e^\top K[:, S]\, e_S$ is exactly its instantaneous decrease; integrating gives the error the run ends at, $R(S, t) = \|e^0 - K[:, S]\, g_t(K_S)\, e^0_S\|^2$ with $g_t(K_S) = K_S^{-1}(I - e^{-tK_S})$. What is ours is the expansion in $t$, and it *supplies* both conditions rather than assuming them. Its $O(t)$ term is exactly $-2t$ times $\sum_{j \in S} e_j (Ke)_j$. **The pointwise criteria of this literature are the first-order truncation of the terminal reward**, modular where it is not, and what they omit grows with the horizon. The filter leaves a fraction $e^{-t\lambda}$ of each mode, so residual mass at $\lambda t \lesssim 1$ survives the horizon whatever is selected, which is what "error the network cannot reduce" means. With no network in it the form bounds nothing: on some instances the pointwise-optimal selection is worse than the best $B$ points by any factor one cares to name (Appendix C).

**The rules are not failing to optimise.** Once the residual has localised, widening the candidate set from $V/B = 1.25$ to 30 buys PINNACLE $405\%$ more of the one-step decrease on `poisson1d` and GRAD-MATCH $73\%$ more, while the trained network is worse, over eight seeds and twenty-four draws per cell. **The sign depends on task and checkpoint both**, $-95\%$ at initialisation on `poisson1d` against $+31\%$ on `crenel`. Eight seeds is few, so we read only cells holding their sign: those two and the $405\%$ do, at $8/8$ and $p \leq 0.008$, while `crenel` at step 50, where the figure is $-40\%$, reaches only $p = 0.20$. The levels are ratios of per-seed medians and spread widely; only the signs are measurements (Appendix J).

**Concentration is the correlate, and it grows as the error localises.** *Concentration* is the mean quantile of $|e|$ over the selected points, *spread* their mean nearest-neighbour distance, normalised so uniform gives 0.5 and 1. On `poisson1d` at step 50 PINNACLE's concentration goes $0.55 \to 0.85$ and its spread collapses to 0.61, while **GRAD-MATCH's spread rises** ($1.01 \to 1.05$) where every pointwise rule's falls. Spread orders the four rules as the exact reverse of concentration on both tasks (Appendix U prints all four); the one-step decrease does not, which is this section's divergence.

**Concentration alone is not the damage**: on a binary MNIST pair the pointwise rules concentrate as hard at no measured cost, $-0.003 \pm 0.013$ over 16 paired seeds, so the chain is concentration *and* a problem where lost coverage is expensive (Appendix U). It is a correlate, and holding it fixed does not remove the damage (below).

**The second condition is a knob, and PINNACLE's standing against uniform follows it.** A `tanh` network cannot represent a jump, so `crenel`'s residual there is irreducible. Three ladders turn that knob independently at 32 seeds and $V/B = 30$ (Table 2), the third not ours, and in each PINNACLE's standing against uniform improves from the sharp end to the broad end. Two cross from losing to winning and `spike` from about even; two of the three reach $p < 0.05$ on the conservative pairing and `spike` alone is unregistered, its amendment having been committed after its runs. GRAD-MATCH improves with the knob too on the one ladder it is run: these separate the knob from the task, not one rule from another.

Table 2: Three ladders on one knob and a fourth that runs the other way; each row gives the seeds of 32 on which PINNACLE beats uniform at $V/B = 30$. A pair of counts means two rungs on that side, and $p$ takes the most conservative pairing the row allows. In the last row "broader" is the larger network (Appendix AB).

| ladder | knob | sharper | broader | $p$ |
|---|---|---|---|---|
| `crenelw` | smoothed jump, half-width $w$ | $3/32$ | $29/32$ | $2.7 \times 10^{-11}$ |
| `spike` | Gaussian bump width $s$ | $17/32, 14/32$ | $30/32$ | $4.4 \times 10^{-4}$ |
| `vburgers1d` | front width $\delta$, *not ours* | $5/32, 9/32$ | $17/32, 25/32$ | $0.074$ |
| network capacity | $2 \times 64 \to 4 \times 64$ on `crenelw` | $17/32$ | $2/32$ | $6.8 \times 10^{-5}$ |

Holding concentration fixed instead of $w$ gives the same direction ($11/32$ against $29/32$, $p = 5.4 \times 10^{-6}$), so **what the criterion must not find is error the network cannot reduce**. Varying the pool size fourfold at a fixed jump width moves nothing ($13/32$, $17/32$, $16/32$), while **capacity beyond $2 \times 64$ strengthens the effect** and reverses the usual objection. The table's last row is depth, and the four-point ladder behind it, $2 \times 32$, $2 \times 64$, $2 \times 128$ and $4 \times 64$, reads 15, 17, 9 and 2 of 32: width does the same and the first step goes the other way, and GRAD-MATCH's own capacity ladder moves only $32/32$ to $28/32$ there, though it is not immune at $4 \times 128$ (Appendix AB).

**And it happens late, which is where the derivation places it.** Window a rule to $V = 100$ for ten of its forty-one selections. At the *end* PINNACLE goes from $30/64$ failures to $2/64$ and BADGE from $18/64$ to $3/64$; at the *start* $32/64$ and $14/64$, which is nothing and four seeds. The direction is not new (Schaul et al., 2016), and Kirsch et al. (2023)'s opposite asymmetry is no contradiction: our damage sits where staleness is *smallest* (Figure 3, and Appendix W.3).

**Our own selector is on the same curve** and degrades worst of all on `poisson1d`, of the five on Appendix K's ladder; on the other four it patterns with the set-level rule and never crosses below uniform, and on `burgers1d` both of our arms cross below the control (Appendix M). This section's claim holds on our own method, where an ablation separates the carried-over fraction of the budget and not the window, and inside RAD's density floor (Appendices D, U).

## 6 Related work

**The phenomenon is *oversearching*, and it is thirty years old.** Quinlan & Cameron-Jones (1995) named it and measured it: a beam width swept from 1 to 512 selected better theories in only one of twelve domains, the cause being a "fluke", and the criterion cannot see it, always improving with further search: Section 5's divergence in 1995. Jensen & Cohen (2000) supplied the statistics: a proof that the bias grows in the number of items examined, and that scores over different $n$ are "incommensurable", which is what our reporting norm rests on. **What is left for us is the instance**:

a fixed *training-point* budget, in a literature reporting neither the width it searches nor a re-drawn control.

**What else is not ours, named.** Neither the phenomenon nor the mechanism is ours; the instance is. Appendix V quotes both and assigns one bullet each to twenty-seven more. **The failure regime itself is Ayed & Hayou (2024)'s**: their Proposition 3 is Proposition 3 in a different limit, theirs the pool growing at a fixed retained fraction, ours the set at a fixed budget.

**The terminal error has a closed form, and it is not ours.** The finite-time predictor for a linearised network trained on $S$ is Jacot et al. (2018)'s, restated by Mohamadi et al. (2022). Lau et al. (2024) write it for PINN collocation, and read it as the quantity selection should maximise (their Eq. 9), in a paper we run as a baseline. What we add is the expansion in $t$ (Appendix C).

**Adaptive collocation for PINNs.** Residual-based refinement adds points where the residual is large (Lu et al., 2021; Nabian et al., 2021); RAD resamples from $p \propto \epsilon^k / \overline{\epsilon^k} + c$ (Wu et al., 2023); R3 is pool-free (Daw et al., 2023). Only two state the size of the set their rule ranks, 100k to 1M (Daw et al., 2023) and $4\times$ the budget (Lau et al., 2024). **That a greedy collocation rule can lose to uniform here is RAD's finding, not ours:** Wu et al. (2023) report "RAR-G and RAR-D are even worse than the Random sampling" at a very small number of residual points, this regime, at an unreported $V$. GRAD-MATCH (Killamsetty et al., 2021) runs unchanged; BADGE (Ash et al., 2020) grows its set as published (Appendix L); both of our recommendations are in Salloum et al. (2026), the set-level repair older still (Kirsch et al., 2019).

**The axis has been measured elsewhere, and the measurements disagree.** At $290\times$ our pool size and a fixed keep-count, 4 of the 7 methods Ivison et al. (2025) examine drop in performance as the pool grows. Five sweeps outside this literature disagree about whether the knob helps, hurts or costs anything (Coleman et al., 2022; Lesci & Vlachos, 2024; Caron et al., 2019; Tsvigun et al., 2022; Grießhaber et al., 2020); and a batch-acquisition sweep two orders wide reports no degradation, which our test predicts (Appendix W.2).

## 7 Limitations

**Scope.** A fixed budget is a condition, not a setting: nothing here transfers to a procedure that only adds points, and it is a named published regime, not ours (Nguyen et al., 2023). The *sorting* is a $B = 40$ statement and inverts by $B = 80$ (Contribution 2, Appendix X). Finite differences beat every arm (McGreivy & Hakim, 2024); every arm here is the same method, so what is compared is where to put the points and not whether to use a network. **Tasks.** The reversal is three tasks at $B = 40$, two of the crossing arms ours: `crenel` under 24 of 30 analysis choices, plus `poisson1d` and `burgers1d` against a re-drawn control. Of the crossing arms that ship at a fixed budget, `crenel` carries PINNACLE and RAD and `burgers1d` PINNACLE; on `poisson1d`, none (Appendix L). All of them are one- or two-dimensional, and neither Hao et al. (2024)'s PINNACLE nor Takamoto et al. (2022)'s PDEBench is run (Appendix AB). A registered replication failed on three more, its smooth control having a jump of 1.2. Two carry a confound we have not removed: on `burgers1d` no arm reaches the $R^2$ of a predictor that ignores time, and enforcing `poisson1d`'s boundary condition exactly makes the control *worse*, taking that cell outside our admissible band (Appendix O.1). **Scale.** The effect itself is read to $B = 160$ at $N = 20{,}000$ (Contribution 1). A registered cell at $5 \times 100$, $B = 2000$, gated at the field's $15\%$ line, *missed* at $28.4\%$ and at $28.7\%$ with L-BFGS, so no $V$ axis is read there; the limit is the budget, not the optimiser (Appendix M). **Mechanism.** The measurement is open-loop and the derivation is a model: linearised, fixed-kernel, bounding nothing. **Estimator.** The two arms carrying the `crenel` crossing score from a sketch that coarsens as $V$ grows, a rival account of their degradation we have not separated (Appendix I).

## 8 Conclusion

Every *pointwise* rule here keeps a high quantile of the first-order truncation of the error a run ends at, over an unreported candidate set. Searching it harder hurts when the score survives selection, the error is irreducible, and the budget cannot grow. **Report $V$; re-draw the control.**

## AI USE STATEMENT

This paper was written with substantial assistance from a large language model. It covers the task categories the ICLR 2027 policy asks authors to report, grouped by where in the work they arose rather than in the policy's order, which that policy permits.

**Writing and presentation.** The model drafted and revised prose throughout, including this sentence, and proposed the title and the keywords. It wrote the plotting scripts that build every figure from the saved runs. Every claim it wrote was checked against a measurement before it was kept, and the checking is the subject of much of the appendix.

**Retrieval and discovery.** It searched for related work, and two of this paper's central attributions came out of that search after the first draft was written: that the phenomenon is Quinlan & Cameron-Jones (1995)'s *oversearching*, and that the order-statistic identity and its growth in the number of items scored are Jensen & Cohen (2000)'s. A third correction came the same way, that the re-drawn uniform control we had presented as unused is Wu et al. (2023)'s Random-R and Daw et al. (2023)'s dynamic baseline. Every quotation in this paper was verified against a PDF of the cited work held locally; the count of verified quotations is checked by a script on each build.

**Ideation and execution.** The research programme and the selection algorithm our two arms implement are the authors'. Within that programme the model proposed the reframing this paper rests on, that the candidate-set size is the axis worth measuring, turned it into testable predictions, designed the experiments that test them, wrote the harness code and the analysis scripts, ran them, and wrote the readings reported here. Each experiment run as a registered test had its predictions, admissible band and refutation conditions committed to a frozen note before it ran, written by the model, and the notes ship with the supplementary material so that the predictions can be read against the outcomes. Three things in this paper are not registered tests and say so where they are used: the 100-seed head-to-head, the `spike` amendment, and the thirty-specification grid. Two registrations failed their own gates and are reported as failed; one run was void and is reported as void.

**Mathematical claims.** The model formulated and wrote the derivations, and the Lean 4 sources as well. Proposition 5 and the modular identity behind Proposition 1 are machine-checked in Lean 4, with the elaboration output shipped, so those two rest on the compiler rather than on the model's word or ours; the remaining propositions are not, and Appendix AB states what the derivation does and does not bound.

**What the model did not do.** It did not generate data in place of measurement: every number in this paper comes from a training run whose per-seed outputs ship. The collocation points and the reference solutions are computed from each task's definition by the harness, which the model wrote and which ships, so the synthetic tasks are its code and not its invention. No dataset was cleaned or relabelled: the one image task reads the standard MNIST archive and restricts it to two digit classes, balanced, in `src/data.py`, with no other filtering. No survey or interview was conducted, so question formulation and transcription do not arise. It did not choose what to believe: the pre-registrations, the two-sided admissibility rule, the multiplicity families and the failure threshold were fixed in advance and are honoured where they refute us. Translation does not arise; the paper was written in English throughout.

**Responsibility.** We have reviewed all AI-assisted work in this paper. The policy asks authors to say how the model's contributions were verified, and the code is the part where that matters most here, because the model wrote the harness, the reimplementations of every baseline and the analysis scripts. Three checks do not rely on it: an independent audit regenerated sixteen of the seventeen script-generated tables and all four figures from the supplementary archive alone and reproduced the committed files byte for byte; `tools/audit_numbers.py` resolves the failure counts in the text to measured cells on disk and prints the ones it cannot; and the harness refuses to write a run when its worker processes disagree about the configuration they used, which is the check that caught a run of ours whose flag never reached the workers. **What we do not claim** is a line-by-line human re-derivation of each reimplementation against the paper it reimplements. We take full responsibility for its final content, including the text, the claims, the code and the derivations produced with the model's aid.

## ETHICS STATEMENT

No human subjects, no personal data, no released model, and every run is on one CPU laptop. The one consideration specific to this paper is that it names published methods and reports them failing, and that the fairness of doing so rests entirely on our reimplementations rather than on the authors' code. We state the departures we know of: our PINNACLE arm is the $k$-means variant with the pool subsampling removed in one table and kept in the sweep, and Appendix P records that, and the fixed budget we impose on RAR-G, dynamicAL and BADGE, whose published forms grow their set. Every arm is reimplemented in one harness so that no method is advantaged by its own implementation, which is also what makes the comparison ours to answer for. Where a rule performs badly here we report the setting that produces it, not a verdict on the method. PINNACLE's own published ratio of about $4$ is not a rung we ran: it sits between $V/B = 2.5$, where PINNACLE fails $1$ of $64$ on `crenel`, and $V/B = 5$, where it fails $18$. Figure 2 marks the default so that a reader can see which of our rungs the method's own setting lies between, and we do not interpolate between them.

## REPRODUCIBILITY STATEMENT

Seventeen of this paper's tables are generated from the saved runs by script, each naming its generator, and an independent audit regenerated sixteen of them and all four figures from the supplementary archive alone and got the committed files byte for byte, and the archive has since been rebuilt so that the seventeenth is in it too. The other fifteen numeric tables are typed into the source from those scripts' printed output and are not regenerable by one command; two of them, the fixed-window table of Appendix Y and the replication table of Appendix Z, have no generator at all and would have to be recomputed from the per-seed files. `tools/audit_numbers.py` resolves most failure counts in the text to a measured cell on disk, and prints the ones it cannot. Most of those are counts pooled over arms, which it does not index because summing across arms would generate enough combinations to match anything and which `scripts/pool_verdict.py` prints directly; the rest are counts of something other than a failure at $R^2 < 0.5$, being the threshold sweep of Appendix Q and the beats-uniform counts of Table 2, and the tool names the cut each of those matches at instead. The harness records the configuration its *worker* processes used, not the one the launching process held, and refuses to write if the workers disagree with each other: that check is what exposed a run of ours whose configuration flag never reached the workers. It was added late, so it is present in nine of the shipped runs and in none of the ones behind the tables here, which is a limitation of this archive and not of the mechanism. Every baseline is reimplemented inside one harness, so two arms share the network, optimiser, data, seed and, among the rules, the re-selection schedule, and differ only in the rule with three exceptions: the uniform control is a single held draw (Appendix O), our own arms score a $V = 200$ window where the other rules score all $N$ in Table 1 and only there, the ladder of Section 4 windowing every arm at the rung's own $V$, and our own arms carry half the budget between rounds where the published ones rebuild it (Appendix K). The departures from published methods, of which Appendix P counts two for PINNACLE alone and the fixed budget for RAR-G, dynamicAL and BADGE, whose published forms all grow their set, are in Appendix P. The designed tests each had margins, endpoints and refutation conditions committed in a note *before* they ran, and where a prediction failed we say so; the $100$-seed head-to-head carries no such registration. **Those notes ship**, one file per design, so the predictions and refutation conditions can be read against the outcomes rather than taken on our word, including the two that failed their own gates and the one whose first execution was void. Proposition 5 is machine-checked in Lean 4 in full, and the modular identity Proposition 1 rests on is checked separately; `lean/` carries the source, the manifest, the toolchain pin and `lean/AXIOMS.txt`, the verbatim elaboration output showing no `sorry` and only `[propext, Classical.choice, Quot.sound]` for all three theorems; the lakefile points at a local mathlib checkout, so a reader supplies their own and redirects that one line. All runs are on one CPU laptop.

## REFERENCES

Jordan T. Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. In *International Conference on Learning Representations (ICLR)*, 2020. arXiv:1906.03671.

Jordan T. Ash, Surbhi Goel, Akshay Krishnamurthy, and Dipendra Misra. Investigating the role of negatives in contrastive representation learning. In *International Conference on Artificial Intelligence and Statistics*, 2022.

Fadhel Ayed and Soufiane Hayou. Data pruning and neural scaling laws: fundamental limitations of score-based algorithms. *Transactions on Machine Learning Research*, 2024. arXiv:2302.06960.

Samuel J. Bell, Onno P. Kampman, Jesse Dodge, and Neil D. Lawrence. Modeling the machine learning multiverse. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Hans-Georg Beyer and Hans-Paul Schwefel. Evolution strategies: A comprehensive introduction. *Natural Computing*, 1(1):3–52, 2002.

Tobias Blickle and Lothar Thiele. A comparison of selection schemes used in evolutionary algorithms. *Evolutionary Computation*, 4(4):361–394, 1996.

Andrea Bonfanti, Giuseppe Bruno, and Cristina Cipriani. The challenges of the nonlinear regime for physics-informed neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. arXiv:2402.03864.

Antoine Bordes, Seyda Ertekin, Jason Weston, and Léon Bottou. Fast kernel classifiers with online and active learning. *Journal of Machine Learning Research*, 6:1579–1619, 2005.

Christos Boutsidis, Michael W. Mahoney, and Petros Drineas. An improved approximation algorithm for the column subset selection problem. In *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pp. 968–977, 2009.

Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. In *International Conference on Learning Representations*, 2019.

Sascha Caron, Tom Heskes, Sydney Otten, and Bob Stienen. Constraining the parameters of high-dimensional models with active learning. *The European Physical Journal C*, 79(11):944, 2019.

Yifan Chen, Ethan N. Epperly, Joel A. Tropp, and Robert J. Webber. Randomly pivoted Cholesky: Practical approximation of a kernel matrix with few entry evaluations. *Communications on Pure and Applied Mathematics*, 78, 2025. arXiv:2207.06503.

Eldan Cohen and Christopher Beck. Empirical analysis of beam search performance degradation in neural sequence models. In *International Conference on Machine Learning*, pp. 1290–1299, 2019.

Cody Coleman, Edward Chou, Julian Katz-Samuels, Sean Culatana, Peter Bailis, Alexander C. Berg, Robert Nowak, Roshan Sumbaly, Matei Zaharia, and I. Zeki Yalniz. Similarity search for efficient active learning and search of rare concepts. In *AAAI Conference on Artificial Intelligence*, 2022. arXiv:2007.00077.

Arka Daw, Jie Bu, Sifan Wang, Paris Perdikaris, and Anuj Karpatne. Mitigating propagation failures in physics-informed neural networks using retain-resample-release (R3) sampling. In *International Conference on Machine Learning (ICML)*, 2023. arXiv:2207.02338.

Pieter-Tjerk de Boer, Dirk P. Kroese, Shie Mannor, and Reuven Y. Rubinstein. A tutorial on the cross-entropy method. *Annals of Operations Research*, 134(1):19–67, 2005.

Yang Fan, Fei Tian, Tao Qin, Xiang-Yang Li, and Tie-Yan Liu. Learning to teach. In *International Conference on Learning Representations (ICLR)*, 2018.

Jose Florido, He Wang, Amirul Khan, and Peter K. Jimack. Investigating guiding information for adaptive collocation point sampling in PINNs. In *Computational Science – ICCS 2024, Lecture Notes in Computer Science*. Springer, 2024. Crossref doi:10.1007/978-3-031-63759-9_36; the arXiv version is arXiv:2404.12282.

Mélanie Gaillochet, Christian Desrosiers, and Hervé Lombaert. Active learning for medical image segmentation with stochastic batches. *Medical Image Analysis*, 90:102958, 2023.

Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. In *International Conference on Machine Learning (ICML)*, pp. 10835–10866, 2023.

Andrew Gelman and Eric Loken. The garden of forking paths: Why multiple comparisons can be a problem, even when there is no "fishing expedition" or "$p$-hacking" and the research hypothesis was posited ahead of time. *Department of Statistics, Columbia University*, 2013. Working paper, 14 November 2013.

Daniel Golovin and Andreas Krause. Adaptive submodularity: Theory and applications in active learning and stochastic optimization. *Journal of Artificial Intelligence Research*, 42:427–486, 2011.

Daniel Grießhaber, Johannes Maucher, and Ngoc Thang Vu. Fine-tuning BERT for low-resource natural language understanding via active learning. In *International Conference on Computational Linguistics (COLING)*, 2020. arXiv:2012.02462.

Nikolaus Hansen and Andreas Ostermeier. Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation*, 9(2):159–195, 2001.

Zhongkai Hao, Jiachen Yao, Chang Su, Hang Su, Ziao Wang, Fanzhi Lu, Zeyu Xia, Yichi Zhang, Songming Liu, Lu Lu, and Jun Zhu. PINNacle: A comprehensive benchmark of physics-informed neural networks for solving PDEs. In *Advances in Neural Information Processing Systems 37 (NeurIPS)*, 2024. Crossref doi:10.52202/079017-2442; the arXiv version is arXiv:2306.08827 (2023).

G. H. Hardy, J. E. Littlewood, and G. Pólya. *Inequalities*. Cambridge University Press, 1934.

Apivich Hemachandra, Gregory Kang Ruey Lau, See-Kiong Ng, and Bryan Kian Hsiang Low. PIED: Physics-informed experimental design for inverse problems. In *International Conference on Learning Representations (ICLR)*, 2025. arXiv:2503.07070.

David Holzmüller, Viktor Zaverkin, Johannes Kästner, and Ingo Steinwart. A framework and benchmark for deep batch active learning for regression. *Journal of Machine Learning Research*, 24(164):1–81, 2023. arXiv:2203.09410.

Daphne Ippolito, Reno Kriz, Maria Kustikova, João Sedoc, and Chris Callison-Burch. Comparison of diverse decoding methods from conditional language models. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 3752–3762, 2019.

Hamish Ivison, Muru Zhang, Faeze Brahman, Pang Wei Koh, and Pradeep Dasigi. Large-scale data selection for instruction tuning. *arXiv preprint arXiv:2503.01807*, 2025. arXiv:2503.01807.

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, 2018. arXiv:1806.07572.

David D. Jensen and Paul R. Cohen. Multiple comparisons in induction algorithms. *Machine Learning*, 38(3):309–338, 2000. doi: 10.1023/A:1007631014630.

Angela H. Jiang, Daniel L.-K. Wong, Giulio Zhou, David G. Andersen, Jeffrey Dean, Gregory R. Ganger, Gauri Joshi, Michael Kaminsky, Michael Kozuch, Zachary C. Lipton, and Padmanabhan Pillai. Accelerating deep learning by focusing on the biggest losers. *arXiv preprint arXiv:1910.00762*, 2019.

Shali Jiang, Daniel R. Jiang, Maximilian Balandat, Brian Karrer, Jacob R. Gardner, and Roman Garnett. Efficient nonmyopic Bayesian optimization via one-shot multi-step trees. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. arXiv:2006.15779.

Jacek Karwowski, Oliver Hayman, Xingjian Bai, Klaus Kiendlhofer, Charlie Griffin, and Joar Skalse. Goodhart's law in reinforcement learning. In *International Conference on Learning Representations*, 2024.

Krishnateja Killamsetty, Durga Sivasubramanian, Baharan Mirzasoleiman, Ganesh Ramakrishnan, Abir De, and Rishabh Iyer. GRAD-MATCH: Gradient matching based data subset selection for efficient deep model training. In *International Conference on Machine Learning (ICML)*, 2021. arXiv:2103.00123.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences (PNAS)*, 114(13):3521–3526, 2017. doi:10.1073/pnas.1611835114.

Andreas Kirsch, Joost van Amersfoort, and Yarin Gal. BatchBALD: Efficient and diverse batch acquisition for deep Bayesian active learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. arXiv:1906.08158.

Andreas Kirsch, Sebastian Farquhar, Parmida Atighehchian, Andrew Jesson, Frederic Branchaud-Charron, and Yarin Gal. Stochastic batch acquisition: A simple baseline for deep active learning. *Transactions on Machine Learning Research*, 2023.

Philipp Koehn and Rebecca Knowles. Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pp. 28–39, 2017.

Andreas Krause, Ajit Singh, and Carlos Guestrin. Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies. *Journal of Machine Learning Research*, 9:235–284, 2008.

Aditi S. Krishnapriyan, Amir Gholami, Shandian Zhe, Robert M. Kirby, and Michael W. Mahoney. Characterizing possible failure modes in physics-informed neural networks. In *Advances in Neural Information Processing Systems*, 2021.

Gregory Kang Ruey Lau, Apivich Hemachandra, See-Kiong Ng, and Bryan Kian Hsiang Low. PINNACLE: PINN adaptive collocation and experimental points selection. In *International Conference on Learning Representations (ICLR)*, 2024. arXiv:2404.07662.

Pietro Lesci and Andreas Vlachos. AnchorAL: Computationally efficient active learning for large and imbalanced datasets. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2024. arXiv:2404.05623.

Ilya Loshchilov and Frank Hutter. Online batch selection for faster training of neural networks. *arXiv preprint arXiv:1511.06343*, 2015. arXiv:1511.06343.

Lu Lu, Xuhui Meng, Zhiping Mao, and George Em Karniadakis. DeepXDE: A deep learning library for solving differential equations. *SIAM Review*, 63(1):208–228, 2021. arXiv:1907.04502.

Michael W. Mahoney. Randomized algorithms for matrices and data. *Foundations and Trends in Machine Learning*, 3(2):123–224, 2011. arXiv:1104.5557.

Levi D. McClenny and Ulisses M. Braga-Neto. Self-adaptive physics-informed neural networks using a soft attention mechanism. *Journal of Computational Physics*, 474:111722, 2023. doi:10.1016/j.jcp.2022.111722; arXiv:2009.04544.

Nick McGreivy and Ammar Hakim. Weak baselines and reporting biases lead to overoptimism in machine learning for fluid-related partial differential equations. *Nature Machine Intelligence*, 6 (10):1256–1269, 2024.

Brad L. Miller and David E. Goldberg. Genetic algorithms, tournament selection, and the effects of noise. *Complex Systems*, 9(3):193–212, 1995.

Sören Mindermann, Jan M. Brauner, Muhammed T. Razzak, Mrinank Sharma, Andreas Kirsch, Winnie Xu, Benedikt Höltgen, Aidan N. Gomez, Adrien Morisot, Sebastian Farquhar, and Yarin Gal. Prioritized training on points that are learnable, worth learning, and not yet learnt. In *International Conference on Machine Learning (ICML)*, 2022. arXiv:2206.07137.

Baharan Mirzasoleiman, Ashwinkumar Badanidiyuru, Amin Karbasi, Jan Vondrák, and Andreas Krause. Lazier than lazy greedy. In *AAAI Conference on Artificial Intelligence*, pp. 1812–1818, 2015.

Jovana Mitrovic, Brian McWilliams, and Melanie Rey. Less can be more in contrastive learning. In *"I Can't Believe It's Not Better!" Workshop at NeurIPS 2020*, 2020.

Mohamad Amin Mohamadi, Wonho Bae, and Danica J. Sutherland. Making look-ahead active learning strategies feasible with neural tangent kernels. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. arXiv:2206.12569.

Kenton Murray and David Chiang. Correcting length bias in neural machine translation. In *Proceedings of the Third Conference on Machine Translation*, pp. 212–223, 2018.

Mohammad Amin Nabian, Rini Jasmine Gladstone, and Hadi Meidani. Efficient training of physics-informed neural networks via importance sampling. *Computer-Aided Civil and Infrastructure Engineering*, 36(8):962–977, 2021. doi:10.1111/mice.12685.

Thi Nguyen Khoa Nguyen, Thibault Dairay, Raphaël Meunier, Christophe Millet, and Mathilde Mougeot. Fixed-budget online adaptive learning for physics-informed neural networks. towards parameterized problem inference. In *International Conference on Computational Science (ICCS)*, 2023.

Myle Ott, Michael Auli, David Grangier, and Marc'Aurelio Ranzato. Analyzing uncertainty in neural machine translation. In *International Conference on Machine Learning*, pp. 3956–3965, 2018.

Deepak Pathak, Pulkit Agrawal, Alexei A. Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *International Conference on Machine Learning*, 2017.

Mansheej Paul, Surya Ganguli, and Gintare Karolina Dziugaite. Deep learning on a data diet: Finding important examples early in training. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

J. Ross Quinlan and R. Mike Cameron-Jones. Oversearching and layered search in empirical learning. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 1019–1024, 1995.

Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019. doi:10.1016/j.jcp.2018.10.045.

Franz M. Rohrhofer, Stefan Posch, Clemens Gößnitzer, and Bernhard C. Geiger. On the role of fixed points of dynamical systems in training physics-informed neural networks. *Transactions on Machine Learning Research*, 2023. arXiv:2203.13648.

David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy P. Lillicrap, and Gregory Wayne. Experience replay for continual learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. NeurIPS 2019.

Hadi Salloum, Maximilian Mifsud Bonici, Sinan Ibrahim, Pavel Osinenko, and Alexei Kornaev. Diversity-aware adaptive collocation for physics-informed neural networks via sparse QUBO optimization and hybrid coresets, 2026. ICLR workshop. arXiv:2603.06761.

Gabriele Santin and Bernard Haasdonk. Convergence rate of the data-independent P-greedy algorithm in kernel-based approximation. *Dolomites Research Notes on Approximation*, 10:68–78, 2017. arXiv:1612.02672.

Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. Prioritized experience replay. In *International Conference on Learning Representations (ICLR)*, 2016.

Bernhard Schölkopf and Alexander J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond.* MIT Press, Cambridge, MA, 2002.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. FaceNet: A unified embedding for face recognition and clustering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. arXiv:1503.03832.

Raphael Silberzahn, Eric L. Uhlmann, Daniel P. Martin, et al. Many analysts, one data set: Making transparent how variations in analytic choices affect results. *Advances in Methods and Practices in Psychological Science*, 1(3):337–356, 2018. doi:10.1177/2515245917747646.

Uri Simonsohn, Joseph P. Simmons, and Leif D. Nelson. Specification curve analysis. *Nature Human Behaviour*, 4:1208–1214, 2020. doi: 10.1038/s41562-020-0912-z.

Jan Simson, Florian Pfisterer, and Christoph Kern. One model many scores: Using multiverse analysis to prevent fairness hacking and evaluate the influence of model design decisions. In *Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency (FAccT)*, 2024. doi: 10.1145/3630106.3658974.

James E. Smith and Robert L. Winkler. The optimizer's curse: Skepticism and postdecision surprise in decision analysis. *Management Science*, 52(3):311–322, 2006.

Ben Sorscher, Robert Geirhos, Shashank Shekhar, Surya Ganguli, and Ari S. Morcos. Beyond neural scaling laws: beating power law scaling via data pruning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Felix Stahlberg and Bill Byrne. On NMT search errors and model errors: Cat got your tongue? In *Proceedings of EMNLP-IJCNLP*, pp. 3356–3362, 2019.

Sara Steegen, Francis Tuerlinckx, Andrew Gelman, and Wolf Vanpaemel. Increasing transparency through a multiverse analysis. *Perspectives on Psychological Science*, 11(5):702–712, 2016. doi: 10.1177/1745691616658637.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F. Christiano. Learning to summarize with human feedback. In *Advances in Neural Information Processing Systems*, 2020.

Makoto Takamoto, Timothy Praditia, Raphael Leiteritz, Dan MacKinlay, Francesco Alesiani, Dirk Pflüger, and Mathias Niepert. PDEBench: An extensive benchmark for scientific machine learning. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2022. doi:10.52202/068431-0117; arXiv:2210.07182.

Kejun Tang, Xiaoliang Wan, and Chao Yang. DAS-PINNs: A deep adaptive sampling method for solving high-dimensional partial differential equations. *Journal of Computational Physics*, 476: 111868, 2023. doi:10.1016/j.jcp.2022.111868; arXiv:2112.14038.

Akim Tsvigun, Artem Shelmanov, Gleb Kuzmin, Leonid Sanochkin, Daniil Larionov, Gleb Gusev, Manvel Avetisian, and Leonid Zhukov. Towards computationally feasible deep active learning. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pp. 1198–1218, 2022.

Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: Lessons learned from the 2015 MSCOCO image captioning challenge. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4):652–663, 2017.

Coen Visser, Alexander Heinlein, and Bianca Giovanardi. PACMANN: Point adaptive collocation method for artificial neural networks. *Computer Methods in Applied Mechanics and Engineering*, 449:118723, 2026. doi:10.1016/j.cma.2025.118723; arXiv:2411.19632.

Haonan Wang, Wei Huang, Ziwei Wu, Hanghang Tong, Andrew J. Margenot, and Jingrui He. Deep active learning by leveraging training dynamics. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022a. arXiv:2110.08611.

Sifan Wang, Xinling Yu, and Paris Perdikaris. When and why PINNs fail to train: A neural tangent kernel perspective. *Journal of Computational Physics*, 449:110768, 2022b. arXiv:2007.14527.

Tingwu Wang and Jimmy Ba. Exploring model-based planning with policy networks. In *International Conference on Learning Representations*, 2020.

Tizian Wenzel, Gabriele Santin, and Bernard Haasdonk. A novel class of stabilized greedy kernel approximation algorithms: Convergence, stability and uniform point distribution. *Journal of Approximation Theory*, 262:105508, 2021. arXiv:1911.04352.

Tizian Wenzel, Gabriele Santin, and Bernard Haasdonk. Analysis of target data-dependent greedy kernel algorithms: Convergence rates for $f$-, $f \cdot p$- and $f/p$-greedy. *Constructive Approximation*, 57:45–74, 2023. arXiv:2105.07411.

Christopher K. I. Williams and Matthias Seeger. Using the Nyström method to speed up kernel machines. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 13, 2001. NeurIPS 2000.

James T. Wilson, Frank Hutter, and Marc Peter Deisenroth. Maximizing acquisition functions for Bayesian optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018. arXiv:1805.10196.

Chenxi Wu, Min Zhu, Qinyang Tan, Yadhu Kartha, and Lu Lu. A comprehensive study of non-adaptive and residual-based adaptive sampling for physics-informed neural networks. *Computer Methods in Applied Mechanics and Engineering*, 403:115671, 2023. arXiv:2207.10289.

Haizhong Zheng, Rui Liu, Fan Lai, and Atul Prakash. Coverage-centric coreset selection for high pruning rates. In *International Conference on Learning Representations*, 2023.

# A ROADMAP: WHICH CLAIM RESTS ON WHICH EVIDENCE

**If you are checking one thing, it is here.** A reviewer with eight papers and one evening should not have to find the right appendix out of twenty-eight. These are the objections this work has actually drawn, each against the appendix that answers it.

| the objection | where it is answered |
|---|---|
| Everything is at $B = 40$; does it survive a real budget? | Appendix X ($B$ to 400 at $N = 20{,}000$), and Appendix M, where a registered cell at $B = 2{,}000$ *missed* its gate |
| You chose the uniform control that flatters you. | Appendix O runs both and says which verdicts move |
| That compute would buy more points instead. | Appendix N, and it beats four of six arms |
| Are these the published rules, or your versions of them? | Appendix P, one row per departure |
| The $R^2 < 0.5$ threshold is doing the work. | Appendices Q and J, four thresholds $\times$ two statistics $\times$ two corrections $\times$ two seed halves |
| It is the churn from re-selecting, not the candidate set. | Appendix Y, five of six cells null |
| What did you try that did not work? | Appendices Z and AB |
| Which parts of this are not yours? | Appendix V, twenty-seven claims with an owner each |

"Prereg." marks the experiments whose margins, endpoints and refutation conditions were written to a design note before the run. One row is marked otherwise: a timestamp audit of `notes/` found that the `spike` amendment was committed two to five minutes *after* the runs it names. We cannot show it was written after the outcomes were read, and we cannot show it was not, so it does not count as registered and the note records why.

| claim | tasks | seeds | prereg. |
|---|---|---|---|
| a larger candidate set makes pointwise rules worse (§4) | `poisson1d`, `crenel` | 64 shared | yes |
| the sign reverses against uniform selection (§4) | `crenel`, `poisson1d`, `burgers1d`; three of six | 64 and 100 | yes |
| the crossing order follows the coverage device (§D) | `crenel` | 64 shared | no |
| more candidates buy more one-step decrease (§5) | `poisson1d`, `crenel` | 8 models $\times$ 24 draws | yes, and P2 was refuted |
| the damage is late, not early (§5, Fig. 3) | `crenel` | 64 | yes, P7 refuted then P8 |
| RAD's floor $c$ reproduces the range (§5) | `crenel`, `poisson1d` | 64 | yes |
| pool size buys nothing (§T) | `crenel` decides; `burgers1d:0.05` underpowered by design | 32 | yes |
| the second condition: the damage goes away as the target is smoothed (§5) | `crenelw`, `spike`, `vburgers1d` | 32 each | `crenelw` and `vburgers1d` yes; `spike`'s amendment was committed after its runs and we do not count it |
| the effect strengthens with capacity (§5) | the smoothed family | 32 | yes |
| the dividing line and the unbounded loss (§C) | derivation, no task | – | Props 1–5; Prop 5 machine-checked in full, and the modular identity Prop 1 rests on |
| the mechanism is the quantile, not the per-round churn (§Y) | `crenel`, `poisson1d` | 64 shared | yes; five of six cells null |
| the effect survives where uniform never fails, and the set-level rule stops being immune (§7, §X) | `crenel`, `poisson1d` at $B = 80, 160$ | 32 shared | yes; the immunity claim was refuted by it |
| the reversal survives a control redrawn on the rules' own schedule (§4, §O) | `crenel` | 64 shared | the experiment was registered, the outcome was not; the rungs held. The $V = B$ anchor carried a registered outcome and it failed on its band |
| the rule the identity is exact for, on the axis (§L) | `crenel`, `poisson1d` | 64 shared | yes, P-ET1–3, all three held |
| the same ladder on the canonical Burgers instance, at our budget (§M) | `burgers1d`, $\nu = 0.01/\pi$ | 64 shared | yes, and the registered prediction that nothing would separate is refuted |
| head-to-head, failure rates (§3) | five tasks | 100 | no |
| replay is the component that matters (§5) | ten tasks pooled, stratified | 5 each, 100 runs | no |
| selection cost (§3) | analytic model plus measured wall clock | – | – |

**What Section 2's count of eleven leaves out.** Two more carry main-text claims: `crenelw` and `spike`, the smoothed families of Section 5's three ladders, which are separate generators in the harness rather than parameterisations of `crenel`. A third, `mnist:3v5`, carries one main-text sentence, in Section 5, and it is a null; so the count of tasks behind a main-text claim is *fourteen*, which is the number Section 2 means by "eleven" plus these three. Those three and two more appear in appendices only: `mnist:3v5` again in Appendices U, Z and AB; `poisson2d` (the ring) and `transport1d` in Appendix O's two-sided void table. Two further MNIST pairs, `mnist:4v9` and `mnist:7v1`, carry no result of their own and enter only pooled, inside the ten-task denominator of Appendix R's ablation, so a claim there does rest on them.

Eleven registered predictions failed and are reported as failures rather than removed: that the published rules would be flat in $V$ (§4), that the one-step gain would hold at every checkpoint (§5), that windowing a prefix of the schedule would recover the benefit (§5), that the concentration a larger pool buys would not grow with training off the PDE (§U), the replication of the crossing itself (§Z), and that the variable behind GRAD-MATCH's exposure was the pool density rather than the budget (§X), and that no rule would sit more than $0.02$ from the re-drawn control at $V = B$, which fails at $0.025$ on a band registered at $1.2$ standard errors (§C), and that no arm would separate from uniform at any rung on `burgers1d` at $\nu = 0.01/\pi$, where five of seven cross below the control (§M). A ninth, that the effect could be reached by varying $B$ instead of $V$, turned out not to be testable and is reported as untested. And one prediction of ours was refuted by our own later run rather than by a registration: that a set-level objective is immune to $V$, which holds at $B = 40$ and fails at $B \geq 80$. **Three more since:** that the early stop and the unmodified rule would be indistinguishable at $B = 40$, refuted by its band on one of four cells (Appendix X); and the two incompatible predictions about `allencahn1d`, the mechanism's and the suppression defence's, both refuted by their letters on a ladder that is a null (Appendix AB). That makes eleven, and this list is a count rather than a curated set: two more registered halves are reported as refuted where they are used, R3's second prediction and P-R1d, and are not repeated here.

## B THE LOSS-DECREASE SCORE AND THE COST MODEL

**The score.** Write $\mathcal{L}(\theta) = \frac{1}{2}\|e(\theta)\|^2$ for the loss on the pool, with $e_j$ the residual at point $j$ (for a PINN, the PDE residual at an interior point or the boundary defect at a boundary point). Under gradient flow $\dot\theta = -\nabla_\theta \mathcal{L}$, the chain rule gives $\dot e = J\dot\theta = -JJ^\top e = -Ke$ with $J = \partial e/\partial\theta$ and $K = JJ^\top$ the empirical NTK (Jacot et al., 2018). Hence

$$-\dot{\mathcal{L}} \;=\; -e^\top \dot e \;=\; e^\top K e \;=\; \sum_j e_j (Ke)_j, \tag{1}$$

so (1) makes $s_j = e_j(Ke)_j$ point $j$'s additive share of the instantaneous loss decrease. Training on a subset $S$ replaces $K$ by $K[:,S]K_S^{-1}K[S,:]$-like terms; we do not use that refinement, and $s_j$ is used only as a ranking signal.

Two things follow that matter for reading the baselines. Residual rules score $|e_j|$, which ranks as $s_j$ does with $K$ replaced by the identity. Diagonal rules score $e_j^2 K_{jj}$, which is $s_j$ with the off-diagonal dropped. Both discard the term that says whether two candidates propagate error to each other, which is the term deflation (Section 5) is designed to exploit.

**The cost model.** We charge a run

$$\underbrace{T_{\text{steps}} \cdot c_{\text{train}}(B)}_{\text{training}} \;+\; \underbrace{(T_{\text{steps}}/c) \cdot c_{\text{sel}}(\cdot)}_{\text{selection}}, \tag{2}$$

with $T_{\text{steps}}$ optimiser steps, written so because $S$ is the selected subset everywhere else, budget $B$, re-selection every $c$ steps. The training term depends on $B$ and not on $N$, so it is identical across every method in a comparison and cancels from any difference. For the selection term, with $P$ parameters, $V$ candidates scored and $m$ Nyström landmarks: a residual pass costs $O(V)$ forward evaluations; forming tangent feature rows costs $O(VP)$; the Nyström factor costs $O(Vm^2 + m^3)$; a greedy pass of $B$ picks with deflation costs $O(BVm)$. Whole-pool rules substitute $V = N$. The implementation is `src/flops.py` and the table is emitted by `scripts/cost_table.py`; the harness charges every logged run with the same function, so no cost number in this paper was typed by hand.

**Why we also report wall clock.** A FLOP count is architecture-independent and a Python implementation is not. Our selector's arithmetic is many small Nyström factorisations, which vectorise far worse than RAD's single vectorised residual pass, so its measured time is worse relative to RAD than its FLOP count implies.

## C THE TERMINAL ERROR IN THE LINEARISED MODEL, AND WHERE $V$ ENTERS

Appendix B stops at the one-step rate and says the subset refinement is not used. This appendix takes it, because that refinement is what the two conditions of Section 5 are conditions *of*. **The closed form**

**below is not ours, and neither is reading it as the objective of selection.** It is the kernel-regression limit of training and appears with the finite-time factor in Jacot et al. (2018); Mohamadi et al. (2022) restate it and build look-ahead active learning on its $t \to \infty$ limit, where the factor is the identity. Lau et al. (2024) go further and closer: their Eq. (9) is this expression, for PINN collocation, in the eigenbasis of the kernel and over the selected set, with the same $(1 - e^{-t\lambda})/\lambda$ filter; they read it as saying which set to choose, define the RKHS norm of the induced residual change as a *convergence degree* $\alpha(Z)$, and bound generalisation by it. That is the paper we run as a baseline, and the reading we would otherwise have claimed is theirs. What is ours is what neither does: **expanding in $t$ and identifying the first-order coefficient's per-point terms, whose diagonal part is the score this literature deploys**, which turns the form into a test on a rule as implemented instead of an objective to maximise, and then asking what it says about $V$.

**The form.** Write $L = \frac{1}{2}\|e\|^2$ for the pool loss, so that the quantity the main text calls the instantaneous decrease is $-\dot{L} = e^\top K[:, S]\, e_S$ exactly. Under gradient flow restricted to $S$, $\dot{e} = -K[:, S]\, e_S$ gives $\dot{e}_S = -K_S e_S$ with $K_S = K[S, S]$, so $e_S(t) = e^{-tK_S} e_S^0$, and integrating the first equation over $[0, t]$, gives the terminal error

$$e(t) \;=\; e^0 - K[:, S]\, g_t(K_S)\, e_S^0, \qquad g_t(K_S) \;=\; K_S^{-1}\big(I - e^{-tK_S}\big). \tag{3}$$

Write $R(S, t) = \|e(t)\|^2$ for the full-domain error the run ends at: the quantity every method here is ultimately judged on, and the quantity none of them optimises. Three things follow.

**Proposition 1** (The pointwise scores are the first-order truncation)**.** *As $t \to 0$,*

$$R(S,t) \;=\; \|e^0\|^2 - 2t\sum_{j\in S} e_j^0 (Ke^0)_j + t^2 c_2(S) + O(t^3), c_2(S) = e^{0\top}K[:,S]K_S e_S^0 + \|K[:,S]e_S^0\|^2. \tag{4}$$

*The $O(t)$ coefficient is* modular *in $S$ and is $-2$ times the per-point score $s_j = e_j(Ke)_j$ of Appendix B summed over the selection. The $O(t^2)$ coefficient is quadratic in $S$ and is not modular in general.*

*Proof.* The qualification is needed: for diagonal $K$ the $O(t^2)$ coefficient collapses to $2\sum_{j\in S} K_{jj}^2 (e_j^0)^2$, which is modular, so the statement holds through the off-diagonal and not otherwise. $g_t(K_S) = tI - \frac{t^2}{2}K_S + O(t^3)$, so $e(t) = e^0 - tK[:, S]e_S^0 + \frac{t^2}{2}K[:, S]K_S e_S^0 + O(t^3)$; squaring gives the display. For the first coefficient, $K[:, S]e_S^0 = \sum_{j\in S} K_{:,j} e_j^0$, hence $e^{0\top}K[:, S]e_S^0 = \sum_{j\in S}(Ke^0)_j e_j^0$, a sum of per-point terms. The second contains $\|K[:, S]e_S^0\|^2 = \sum_{i,j\in S} e_i^0 e_j^0 (K^2)_{ij}$, which is not; here $(K^2)_{ij} = \sum_{k=1}^N K_{ki}K_{kj}$ is an entry of the square of the *full pool* kernel, not of $K_S$. □

The modular identity is machine-checked: `lean/Selection.lean`, theorem `first_order_modular`, Lean 4 with mathlib, no `sorry` and no added axiom.

So a criterion that scores points by $s_j = e_j(Ke)_j$ agrees with the terminal reward *to first order in the horizon* and may disagree at every higher order, and the neglected terms grow with $t$. A residual or diagonal score is a second truncation inside the first and disagrees already at $O(t)$, by the off-diagonal sum $\sum_{k\neq j} K_{jk}e_k$. That is why the one-step state statistics we tried and refuted were the wrong readings: each is a $t \to 0$ quantity, and at $t \to 0$ the model says $s_j$ is exactly right. It also rules out the obvious repair. Two selected points interact through $c_2$ *even when their kernel entry is exactly zero*, because $(K^2)_{ij}$ couples them through the rest of the pool: the generating script constructs such a pair and measures a modularity defect of $-0.264\, t^2$ at $K_{ij} = 0$. A pairwise diversity term that reads $K_{ij}$ does not see it either.

**Proposition 2** (Irreducibility is a statement about the horizon)**.** *Let $v$ be a unit eigenvector of $K_S$ with eigenvalue $\lambda$. Then $|v^\top e_S(t)| = e^{-t\lambda}\,|v^\top e_S^0| \geq e^{-\eta}\,|v^\top e_S^0|$ whenever $\lambda t \leq \eta$. The bound depends on $S$ only through its spectrum, not on how many points were chosen or which.*

*Proof.* $e_S(t) = e^{-tK_S} e_S^0$, and $e^{-t\lambda}$ is decreasing in $t\lambda$. □

Residual mass in directions with $\lambda t \lesssim 1$ therefore survives the horizon, and **which matrix supplies $\lambda$ decides whether that is a statement about the problem or about the rule**. Read on $K_S$, as it is

stated, the proposition is about a given selection, and a rule that keeps near duplicates manufactures small eigenvalues of $K_S$ by itself: that is precisely what the construction in Proposition 4 does, with $\varepsilon$ as the manufactured eigenvalue. Nothing rule-independent follows from it. Read on the full-pool $K$, which is the same proposition with $S$ the whole pool, the mass in the pool's own slow modes survives the horizon whoever selects, and that is what "error the network cannot reduce" means here. The two are different claims and only the second is a property of the target and the horizon rather than of the rule. The last column of Table 3 measures the second: it is computed from the 400-point grid kernel in `scripts/terminal_model.py` and not from any selected submatrix. Smoothing the target moves mass to larger $\lambda$, which is the intervention behind the $w$ and $\delta$ ladders of Section 5; the last column of Table 3 measures that fraction directly and it runs from $0.62$ to $0.000$ across the width ladder.

**Proposition 3** (A pointwise selection collapses, and its coverage floor does not depend on the budget)**.** *Let $\Omega \subset \mathbb{R}^d$ be compact, $\mu$ a probability measure with $\operatorname{supp}\mu = \Omega$, and $g : \Omega \to \mathbb{R}$ continuous with $M = \max_\Omega g$. For $\eta > 0$ put $A_\eta = \{x : g(x) \geq M - \eta\}$ and $A_0 = \bigcap_{\eta>0} A_\eta$, the set $\{x : g(x) = M\}$ of maximisers. Draw $V$ candidates i.i.d. from $\mu$ and keep the $B$ with the largest $g$, giving $S_V$. Then for fixed $B$ and every $\eta > 0$, $\Pr[S_V \subset A_\eta] \to 1$ as $V \to \infty$, and the fill distance $h(S) = \sup_{x\in\Omega} \min_{j\in S} \|x - x_j\|$ satisfies $\liminf_{V\to\infty} h(S_V) \geq \sup_{x\in\Omega} d(x, A_0)$ in probability. That limit is determined by $g$ and $\Omega$ alone and does not depend on $B$.*

*Proof.* $g$ is continuous and $\operatorname{supp}\mu = \Omega$, so $\mu(A_\eta) > 0$. The number of candidates landing in $A_\eta$ is $\mathrm{Bin}(V, \mu(A_\eta))$, so the probability that fewer than $B$ do tends to 0. Every point of $A_\eta$ scores at least $M - \eta$ and every point outside scores strictly less, so as soon as $B$ candidates lie in $A_\eta$ the kept set lies in it too, whence $h(S_V) \geq \sup_x d(x, A_\eta)$. Finally $A_\eta \downarrow A_0$ and $\Omega$ is compact, so $\sup_x d(x, A_\eta) \to \sup_x d(x, A_0)$. □

Fix the budget $B$ and write $q_V(t)$ for the $1 - B/V$ quantile of the score at time $t$. Keeping the best $B$ of $V$ places every selected point above $q_V(t)$, so the rule is blind to $\{g < q_V(t)\}$, whose score-measure is $1 - B/V$ by construction. Let

$$D(V,T) \;=\; \frac{\text{residual mass at points scoring below } q_V(T), \text{ in modes with } \lambda < 1/\Delta t}{\text{total residual mass}}$$

be the share of the residual that the rule cannot see *and* the next stretch of training will not remove. $\Delta t$ is the interval between re-selections, which is 100 steps at every exam and therefore the same number every time, so the spectral cutoff $1/\Delta t$ is fixed. $T$ is total elapsed training time, which grows.

**The pool-density null, named, because it was not.** Section 5 says varying the pool size fourfold at a fixed jump width moves nothing, $13/32$, $17/32$, $16/32$. Those are `logDU_N600`, `logDT_w0.001` and `logDU_N2400`, all `crenelw:0.001` at 32 seeds, counting seeds on which PINNACLE beats uniform. An independent recompute of this paper could not find them, looked at the width ladder's `logDT_w0.0001*` runs instead, and got 13, 5, 15: the first number agrees by coincidence, which is the worst case for a reader checking by hand.

**$D$ has a zero, the zero is an arm, and the ladder now has an anchor.** At $V = B$ the quantile $q_V(t)$ is the 0 quantile, $\{g < q_V(t)\}$ is empty, and $D(B, t) = 0$ for every $t$, every task and every rule that keeps all $B$ of the $B$ it is shown. The family of curves in Section 4 therefore has a left endpoint that the definition fixes and such a rule cannot move: at $V = B$ it must keep every candidate it was shown, so it *is* a uniform draw of $B$ points, re-drawn on the rule's own schedule. Our own two arms are not such a rule and the endpoint is not fixed for them: they keep $b = B/2$ per round, so at $V = B$ the quantile is the median rather than the zero, $\{g < q_V\}$ is half the window, and $D(B, t) > 0$. They were also never run at $V = B$; the anchor cell below carries the five `win:` arms and the two controls only. The endpoint claim is about the published rules on this ladder, which is what the anchor tests. That arm is the re-drawing control of Appendix O. A held draw is *not* a member of this family at any $V$, which is the sharpest way to say why the two controls are not interchangeable.

**The anchor is measured, and the prediction we registered about it is refuted on its band.** We ran the ladder's missing rung, $V = B = 40$ on `crenel`, 64 seeds, and registered before the run that no `win:` arm would reach $p < 0.05$ against the re-drawn control and that every paired mean would lie inside $\pm 0.02$. The first holds for all five: $p = 0.127$, $0.127$, $0.224$, $0.342$, $0.963$. The second fails for

three, at $+0.0254$, $+0.0254$ and $+0.0224$, so the prediction is refuted and is reported as refuted. It is refuted on a band we set at $1.2$ standard errors of the statistic it bounds, which is a registration error rather than a finding, and a reader should discount it accordingly and not the other way round.

The run does establish what the anchor needs, that no `win:` rule is distinguishable from a uniform re-draw at $V = B$, and reading the implementations afterwards says why more strongly than the argument above: each of these rules takes $B$ of $B$ *without replacement*, so each returns the whole window whatever its score says. The five then differ from one another only in how fast they consume the random stream, and therefore in which window they see at the *next* exam. BADGE and PINNACLE consume it identically and are byte-identical on $64$ of $64$ seeds; that is consistent with the algebra of Appendix P and is not an independent test of it, because at $V = B$ the score does not enter the selection at all.

The anchor buys a decomposition the paper could not make before. At the smallest rung we ran, $V/B = 1.25$, where the truncation discards $20\%$, the five `win:` arms sit $+0.012$ to $+0.041$ above the anchor and three of the five are inside noise; our two arms sit $+0.093$ and $+0.096$, both significant. The `win:` arms differ from the anchor in one thing, how hard the score is optimised, and at $V/B = 1.25$ there is little of that to have. Ours differ in two: they also retain $1 - 1/L$ of the budget across exams (Appendix D).

**What it is not.** $D$ is defined, not estimated: we do not measure it in any run, because the eigendecomposition it needs is the $O(N^3)$ object this literature avoids.

**Proposition 4** (The loss is unbounded, not a constant factor)**.** *For every $B \geq 2$ and every $\rho > 0$ there is a pool, a positive-definite $K$ and a residual $e^0$ on which the pointwise-optimal selection, meaning the top $B$ by $|e^0|$ and hence the $V/B \to \infty$ limit of any rule scoring points individually by residual magnitude, satisfies $R(S_{\mathrm{pw}}) > \rho \cdot \min_{|S|=B} R(S)$, where $R(S) = \lim_{t\to\infty} R(S,t)$.*

*Proof.* Take $N = m + B - 1$ points with $m \geq B$, reals $a > b > 0$ and $\varepsilon > 0$. Let $K$ be block diagonal: $K_1 = J_m + \varepsilon I_m$ on the first $m$, where $J_m = \mathbf{1}_m \mathbf{1}_m^\top$, not the Jacobian of Appendix F, so that block is $m$ near-duplicates; and $K_2 = I_{B-1}$ on the rest. Let $e^0 = a$ on the first block and $b$ on the second. Then $K \succ 0$.

Since $a > b$ the top $B$ by $|e^0|$ lie in the first block. There $(J_B + \varepsilon I_B)\mathbf{1}_B = (B + \varepsilon)\mathbf{1}_B$, so $K[S,S]^{-1} e^0_S = \frac{a}{B+\varepsilon}\mathbf{1}$; each remaining first-block point is predicted at $\frac{aB}{B+\varepsilon}$ and each second-block point at $0$, giving

$$R(S_{\mathrm{pw}}) \;=\; (m-B)\Big(\tfrac{a\varepsilon}{B+\varepsilon}\Big)^2 + (B-1)b^2 \;\geq\; (B-1)b^2. \tag{5}$$

Take instead one first-block point together with all $B-1$ of the second block. The second block is then predicted exactly, and $R(S_\star) = (m-1)\big(\frac{a\varepsilon}{1+\varepsilon}\big)^2$. The ratio is at least $(B-1)b^2(1+\varepsilon)^2/[(m-1)a^2\varepsilon^2]$, which exceeds any $\rho$ once $\varepsilon$ is small enough. □

Both closed forms are checked against the linear solve to $10^{-9}$ in `scripts/terminal_model.py`. Two things the ratio alone does not settle, both measured there as well. It is unbounded because the *denominator* vanishes, which is a weak kind of unbounded; taking $m = B$ so that no first-block point is left over, the pointwise set instead leaves a constant *fraction* of the initial error, $0.44$ of $\|e^0\|^2$ at $B = 8$, where the optimum leaves $8 \times 10^{-8}$. And the statement is about $t \to \infty$, taken before $\varepsilon \to 0$, which is the limit in which $K[S,S]^{-1}$ blows up; its norm is $\Theta(1/\varepsilon)$ and does not depend on $t$. Shrinking $\varepsilon$ at a fixed short horizon does not give an unbounded ratio: `scripts/order_of_limits.py` evaluates the construction at finite $t$ on the same three instances as the table above, and over $\varepsilon$ from $10^{-1}$ to $10^{-6}$ the ratio moves by a factor of $1.1$ to $5.9$ at $t \leq 1$ and by $8 \times 10^9$ at $t \to \infty$. At a fixed finite $t$ it is bounded, because the denominator does not vanish there. The proposition is stated for $R(S) = \lim_{t\to\infty} R(S,t)$, so it takes the limits in the order that makes it true. It also **reverses at short horizon**: at $t = 0.1$ the pointwise set is the better of the two, ratio $0.52$, and by $t = 1$ it is worse, ratio $2.07$. That is Proposition 1 visible in a single instance, since at $t \to 0$ the pointwise score is the criterion the terminal reward actually wants.

**This failure is not new, and the credit is Kirsch et al. (2019)'s.** Kirsch et al. (2019) report that pointwise batch acquisition "acquires similar and redundant points, sometimes performing worse

than randomly acquiring data", and repair it with an objective on the selected batch carrying a $1-1/e$ guarantee. That is a readable proxy for this paper's dividing line, found independently in Bayesian active learning six years earlier.

What the construction above adds is *where the redundancy comes from*. Their Repeated MNIST puts duplicates into the data by hand. Here the pool contains none, and a larger $V$ manufactures them, because by Proposition 3 the top $B$ of $V$ collapses into a shrinking region of the domain, so near-duplicates are what a dense pool looks like once the criterion is optimised hard enough. The failure is theirs; the knob that produces it without any duplicated data is ours. The horizon behaviour also has no counterpart there: at $t = 0.1$ the pointwise set is the better of the two, and only a longer horizon reverses it.

**The complement half, and why the counterexample and the failed replication are one statement.** Split $R(S,t)$ by which points were selected. The part living on $S^c$ is $e^0_{S^c} - K[S^c,S]\, g_t(K_S)\, e^0_S$: the abandoned residual, minus whatever the kernel carries out to it from the selection. The two conditions are statements about what a criterion *picks*. This term is a statement about what it *leaves*, and it is large exactly when $K[S^c,S]$ transfers little. Proposition 4's instance takes it to the limit: $K$ there is block diagonal, so the transfer is identically zero, which is why an unbounded loss is available at all.

**$V$ enters the two cases through different doors.** By Proposition 3, for a pointwise criterion $V$ moves *which set is selected*, and pushes it towards a configuration with a coverage floor that no budget removes. For a rule whose objective is a submodular set function, greedy over a random ground set attains a $(1-1/e-\varepsilon)$ fraction of the optimum, with the shortfall $\varepsilon$ controlled by how many candidates are sampled at each step (Mirzasoleiman et al., 2015). There the candidate count enters only a *lower bound*, and a lower bound does not get worse when the ground set grows. GRAD-MATCH's own guarantee is weaker than that, and has no $V$ in it either (Appendix P).

**Proposition 5** (The residual becomes progressively irreducible)**.** *Let $e_t = e^{-tK}e^0$ with $e^0 = \sum_i c_i v_i$ in an eigenbasis of $K$, and weight the modes by $w_i(t) \propto c_i^2 e^{-2t\lambda_i}$, so that $\sum_i w_i(t) = 1$ and $w(t)$ is how $\|e_t\|^2$ is distributed over the spectrum. Then $t \mapsto \mathbb{E}_{w(t)}[\phi(\lambda)]$ is non-decreasing for every non-increasing $\phi$. In particular the share of the residual carried by modes with $\lambda < \tau$ is non-decreasing in $t$, for every $\tau > 0$.*

*Proof.* $w(t)$ is an exponential family in $t$ with sufficient statistic $-2\lambda$, so it has monotone likelihood ratio in $-\lambda$: for $t' > t$, $w(t')$ stochastically dominates $w(t)$ in that order. The expectation of a non-increasing function of $\lambda$ is therefore non-decreasing. Taking $\phi = \mathbf{1}[\lambda < \tau]$ gives the second statement. □

This one is machine-checked in full: `lean/Selection.lean`, theorem `tilt_monotone`, stated cross-multiplied so that no denominator has to be shown nonzero. Its engine is the weighted Chebyshev sum inequality for similarly ordered families, which is classical (Hardy et al., 1934); it is proved in that file as `weighted_chebyshev` only because mathlib carries the unweighted form alone, which is a fact about the library and not about the inequality. All three report `[propext, Classical.choice, Quot.sound]` under `#print axioms`, which is to say nothing was assumed and nothing was left open.

**This is the timing asymmetry, and it follows from the other three.** By Proposition 2 the mass at $\lambda t \lesssim 1$ is what the network cannot remove within the horizon. Proposition 5 says the share of the residual sitting there only grows. So a rule scoring by $|e|$ reads, later in training, a field that is progressively more concentrated on exactly the error the second condition says is harmful to chase, and by Proposition 3 a larger $V$ collapses the selection onto that field's maximisers. The same $V$ therefore does more damage later, which is what restricting a rule to the last ten of forty-one selections measures and the first ten do not. One thing this does not do is remove the open-loop caveat. The trajectory in Proposition 5 is the full-pool one, $e_t = e^{-tK}e^0$, which is exactly the reference trajectory the mechanism measurement uses and is *not* the trajectory of a run that selects. What is derived is that the score field a rule reads becomes progressively irreducible along the reference run; the timing asymmetry itself is measured in the closed loop, and joining the two inherits the caveat in Section 7. In the model the coverage floor rises from $0.216$ to $0.647$ as the irreducible share goes from $0.002$ to $0.514$; we report that the floor does most of its growing early and then flattens, so the model gives the sign and the ordering of the timing effect and not its shape.

**What the form predicts, against measurements it was not fitted to.** Table 3 evaluates the terminal error $R(S,t)$ on a synthetic instance: an RBF kernel of bandwidth $\ell$ on $[0,1]$ standing in for the NTK, a residual bump of width $w$, a budget of $B = 12$ points from a pool of $N = 400$, and two selected sets: *spread*, evenly spaced, and *collapsed*, the top $B$ of $|e|$, which is the $V/B \to \infty$ limit of any rule that scores the residual pointwise. The generating script `scripts/terminal_model.py` first verifies the closed form against explicit integration of $\dot{e} = -K[:,S]e_S$, agreeing to $1.5 \times 10^{-8}$, and refuses to emit the table otherwise. Three facts this paper reports *from training* appear in a table with no network in it:

1. Selection can end worse than selecting nothing at all (Section 3): the rows marked harmful, where the collapsed set leaves more error than the untouched residual carried.
2. It does so *only* where the residual is sharper than the kernel can propagate, $w/\ell < 1$. At $w/\ell \geq 1$ no row is harmful at any horizon. That is the second condition, and the three ladders of Section 5 are its experimental form. This holds against the no-selection baseline, which is the comparison the word "harmful" names here and the one Section 3 makes when it reports selection ending worse than selecting nothing. Against the *spread* set instead, which is the model's nearest analogue of uniform selection, the pattern is not the second condition. Collapse is worse than spread at a bump-to-bandwidth ratio of $0.1$, better at $0.2$, $0.4$ and $1$, and worse again at $2$ and $4$ ($15.396$ against $0.028$, and $77.542$ against $0.011$, at horizon $t = 10$). The right-hand end of that is a property of the model rather than of the experiments, and it is where the model *disagrees* with them: on the broad rung of the `spike` ladder the pointwise rule wins $30/32$ against uniform, where twelve evenly spaced points in this model are already nearly exact and abandoning them is expensive. A single-selection model with no network in it has no way to represent the training that repairs a thin design, so we read the table against no selection and say here what it does not carry.
3. Collapse is not itself harm. At $w/\ell = 1$ the collapsed set *beats* the spread one, $0.002$ against $1.813$ at $t = 100$, and $0.005$ against $1.814$ at $t = 10$, so the reading does not turn on the horizon. Concentration is necessary and not sufficient, which is what the binary MNIST measurement says with a trained network.

Table 3: $R(S,t)$ in the linearised model, from `scripts/terminal_model.py`. No network is trained. "Harmful" means the collapsed set ends above the no-selection baseline, which is not the same comparison as beating uniform selection; the text says what changes under the other. The last column is the fraction of $\|e^0\|^2$ lying in eigendirections of $K$ with $\lambda t < 1$, the horizon's cutoff.

| $w/\ell$ | no selection | spread $S$ | collapsed $S$ | harmful? | mass at $\lambda t < 1$ |
|---|---|---|---|---|---|
| horizon $t = 10$ | | | | | |
| 0.10 | 3.54 | 3.536 | 4.698 | **yes** | 0.621 |
| 0.20 | 7.07 | 7.071 | 6.815 | no | 0.323 |
| 0.40 | 14.14 | 11.025 | 8.106 | no | 0.048 |
| 1.00 | 35.36 | 1.814 | 0.005 | no | 0.000 |
| 2.00 | 70.72 | 0.028 | 15.396 | no | 0.000 |
| 4.00 | 141.39 | 0.011 | 77.542 | no | 0.000 |
| horizon $t = 100$ | | | | | |
| 0.10 | 3.54 | 3.536 | 366.184 | **yes** | 0.567 |
| 0.20 | 7.07 | 7.071 | 126.271 | **yes** | 0.252 |
| 0.40 | 14.14 | 11.025 | 4.981 | no | 0.022 |
| 1.00 | 35.36 | 1.813 | 0.002 | no | 0.000 |
| 2.00 | 70.72 | 0.028 | 12.016 | no | 0.000 |
| 4.00 | 141.39 | 0.011 | 70.700 | no | 0.000 |
| horizon $t = 1000$ | | | | | |
| 0.10 | 3.54 | 3.536 | 3475.117 | **yes** | 0.515 |
| 0.20 | 7.07 | 7.071 | 1392.862 | **yes** | 0.193 |
| 0.40 | 14.14 | 11.025 | 90.647 | **yes** | 0.009 |
| 1.00 | 35.36 | 1.813 | 0.000 | no | 0.000 |
| 2.00 | 70.72 | 0.028 | 6.640 | no | 0.000 |
| 4.00 | 141.39 | 0.011 | 58.898 | no | 0.000 |

**What the horizon column does not show.** Harm also deepens with $t$, from one harmful row at $t = 10$ to three at $t = 1000$, and it is tempting to read that as the timing asymmetry of Section 5. What the column shows is that harm deepens the more completely the collapsed set is fitted, which is consistent with the damage being late and is not a derivation of it.

**This is also why the capacity row of Table 2 runs the other way from the three smoothness ladders.** Read naively it looks like a counterexample to the second condition. A wider network represents more of the residual, so the second condition read alone predicts *less* damage; what is measured is more. The two are about different halves of the run. The second condition is about what the criterion *finds* before it selects, and it is held fixed along the capacity ladder because the task is. Capacity governs what happens *afterwards*, how completely the training then fits whatever set was chosen, which is the column above. A collapsed set fitted more exactly generalises worse, so capacity should and does deepen the harm. We have not derived the capacity effect and do not claim to; what we claim is that it is not the same variable as the one the ladders move.

**What this is not.** A linearised model with a fixed kernel, gradient flow rather than Adam, a bump rather than a PDE residual, and a hand-placed collapsed set rather than a rule running. It is not evidence about physics-informed networks; the experiments are that. Its use is that the sign and the ordering of the three measured effects listed above follow from one expression that no part of this harness was fitted to, and that a fourth, the broad rung of the `spike` ladder, does not, and that the expression names the object the field's criteria are proxies for. Two predictions it makes that we have *not* tested are recorded in Appendix AB.

## D THE SELECTOR IN FULL

If a rule should be shown few candidates, the design question is *what the best rule that only ever sees $V$ of them looks like*. Ours fixes the score $s_j$ and combines a window $W$ of $V$ candidates re-drawn each round, greedy selection with *deflation* $D$ (Boutsidis et al., 2009), and a first-in first-out *replay tail* $R$ carrying over $1 - 1/L$ of the budget each round (Kirkpatrick et al., 2017; Rolnick et al., 2019). It is not our recommendation (Section 3), and the ablation is what it is here for.

**Only one of the three does measurable work.** The ten tasks are `poisson1d`, `crenel`, `sinjump`, `nlrd1d`, `vburgers1d`, `blobs_boundary`, `burgers1d:0.05` and three MNIST pairs (`3v5`, `4v9`, `7v1`); it is not the eleven of Section 2, and the arithmetic is worth writing out because subtracting two and adding two would leave eleven: it drops `allencahn1d` and `convdiff1d`, which this ablation predates, and `staircase` and `tri_zone`, which it never ran, leaving seven; it adds three MNIST pairs, none of which is among the eleven, to give the continuous outcome more cells; and it reads `burgers1d` at $\nu = 0.05$, which Appendix AB calls a different task. Pooling ten tasks $\times$ two scores $\times$ five seeds gives 100 runs per configuration: all three on, $1/100$ failures ($[0.002, 0.054]$); no window, $5/100$; no replay, $8/100$ $[0.041, 0.150]$. **The replay tail separates and the window does not**, all-on against no-replay is Fisher $p = 0.035$ and $p = 0.027$ stratified by task, surviving the threshold band from $0.25$ to $0.9$ (stratified $p \leq 0.027$). It is a band and not the whole axis: pooled, replay is $p = 0.059$ at a threshold of $0$ and $0.098$ at $0.95$, so the significant band starts one notch above its lowest end and stops one below its highest, and both ends are cells where almost nothing or almost everything fails. The window separates at no threshold pooled ($p \geq 0.059$) and at two of the six thresholds stratified ($p = 0.044$ at $0$ and $0.25$, $p \geq 0.15$ above). We report both because they disagree, and the window's case rests on the weaker of the two. The sweep and both tests are `scripts/crash_rate.py`, which takes the threshold as an argument. That the window does not separate is Section 4's claim holding on our own method.

The score is $s_j = e_j(Ke)_j$ of Appendix B. At each re-selection the rule draws a fresh uniform window of $V$ pool indices; forms the tangent feature rows for those $V$ points only, so no part of the step is $O(N)$; and picks $B/L$ new points greedily, projecting each chosen point's tangent direction out of the remaining candidates after every pick. Without that deflation a budget spent on the top of any magnitude ranking buys $B$ near-duplicates of the single best point; it is the pivoting step of column-subset selection (Boutsidis et al., 2009) applied to tangent features. The selection is held in a first-in-first-out buffer of $L$ blocks: each round admits the new block and evicts the oldest, so a fraction $1 - 1/L$ of the budget is carried over. All experiments use $L = 2$, $V = 200$ unless swept, and re-selection every $c = 100$ steps. Two scores are reported throughout, `addpiv` and `kepiv`,

which differ in how the residual and the kernel term are combined; the ablation of Section 5 pools over both, so its conclusion does not depend on that choice.

Table 4: On `crenel`: the first $V/B$ at which a rule is worse than the *held* uniform draw, which fails $2/64$ throughout, with Fisher $p < 0.05$, beside the coverage device its definition carries. The crossings order the devices *that act on the domain*; BADGE's is diversity in embedding space, which Holzmüller et al. (2023) classify as satisfying diversity and which buys no domain coverage here.

| rule | coverage device in its definition | crosses below uniform at $V/B$ |
|---|---|---|
| PINNACLE | $k$-means++, as BADGE's | 5.0 ($p < 0.001$) |
| BADGE | k-means++, in embedding space | 5.0 ($p = 0.030$) |
| RAD | the floor $c$ in $p \propto \varepsilon^k/\overline{\varepsilon^k} + c$ | 20 ($p = 0.016$) |
| ours | deflation against the already-selected | never |
| GRAD-MATCH | objective defined on the selected set | never; improves monotonically |

## E THE CANDIDATE-SET GRID IN FULL

The failure-rate curve of Figure 2 is plotted from these numbers. Failures at $R^2 < 0.5$ against the candidate-set size, 64 shared seeds per cell, $N = 1{,}200$, $B = 40$. $V/B$ is candidates inspected per point kept, so $V/B = 30$ is the whole pool. `uniform` is a control that cannot see the candidate set and does not move. Generated by `scripts/vcurve.py` from `log{DF,ET}_{crenel,poisson1d}_V{V}`: `logDF` carries the published rules and ours, `logET` the re-drawn control and RAR-G. That one command writes both this table and Figure 2, and re-running it reproduces the two files byte for byte, so the figure and the table cannot have drifted apart.

Table 5: **The candidate-set ladder in full.** Seeds failing at $R^2 < 0.5$ out of the 64 shared per cell, $B = 40$ kept from $N = 1{,}200$, $[64, 64]$ with tanh, 4000 Adam steps, re-selection every 100. $V/B = 30$ is the whole pool. `uniform` is the held draw and cannot see the candidate set. Figure 2 plots these numbers. Generated by `scripts/vcurve.py` from `log{DF,ET}_{crenel,poisson1d}_V{V}`.

| $V$ | $V/B$ | uniform | re-drawn | RAR-G (top-$B$) | RAD | PINNACLE | BADGE | GRAD-MATCH | ours (addpiv) | ours (kepiv) |
|---|---|---|---|---|---|---|---|---|---|---|
| `crenel` | | | | | | | | | | |
| 50 | 1.25 | 2/64 | 0/64 | 0/64 | 0/64 | 0/64 | 0/64 | 1/64 | 0/64 | 0/64 |
| 100 | 2.5 | 2/64 | 0/64 | 64/64 | 3/64 | 1/64 | 1/64 | 0/64 | 0/64 | 0/64 |
| 200 | 5 | 2/64 | 0/64 | 64/64 | 2/64 | 18/64 | 10/64 | 0/64 | 0/64 | 0/64 |
| 800 | 20 | 2/64 | 0/64 | 64/64 | 11/64 | 27/64 | 24/64 | 0/64 | 0/64 | 0/64 |
| 1200 | 30 | 2/64 | 0/64 | 64/64 | 6/64 | 30/64 | 18/64 | 0/64 | 0/64 | 0/64 |
| `poisson1d` | | | | | | | | | | |
| 50 | 1.25 | 35/64 | 1/64 | 1/64 | 4/64 | 2/64 | 2/64 | 2/64 | 1/64 | 2/64 |
| 100 | 2.5 | 35/64 | 1/64 | 23/64 | 3/64 | 6/64 | 5/64 | 0/64 | 0/64 | 2/64 |
| 200 | 5 | 35/64 | 1/64 | 58/64 | 5/64 | 5/64 | 8/64 | 0/64 | 9/64 | 8/64 |
| 800 | 20 | 35/64 | 1/64 | 64/64 | 10/64 | 7/64 | 10/64 | 1/64 | 26/64 | 16/64 |
| 1200 | 30 | 35/64 | 1/64 | 64/64 | 10/64 | 11/64 | 10/64 | 0/64 | 24/64 | 26/64 |

**The same ladder without a threshold.** A failure rate needs a cut, and on `crenel` the outcome is graded rather than two-valued, so the cut is doing work there that the effect size can do without it. Below is the paired mean of (rule − uniform) in test $R^2$ over the same shared seeds, bold where the two-sided Wilcoxon signed-rank $p < 0.05$; it is generated by `scripts/vcurve_paired.py` from `logDF` alone, which is the table above without its `logET` columns: passing both adds a re-drawn row this table does not carry. The sign reversal on `crenel` appears in the effect size, near-monotonically, and RAD's crossing, which is the weakest cell of the count reading, survives it at Holm-corrected $p = 0.015$ at its own rung $V/B = 20$; one rung further out, at $0.059$, it does not. On `poisson1d` the same statistic corrects an impression the ratios give: the published rules degrade but stay $+2.2$ to $+4.6$ above uniform throughout, and the only arm whose *mean* ends below this control is ours, and it is both of them. That last reading is the mean of an unbounded quantity and it does not survive a robust statistic: `addpiv` at $V/B = 30$ has a paired median of $+0.060$ with 37 of 64 seeds positive ($p = 0.54$) and fails $24/64$ against the control's $35/64$, so against the *held*

control it ends below only in the tail. Against the re-drawn control it ends below on the median too ($-0.165$, $p = 5.7 \times 10^{-7}$), which is where that sentence belongs.

Table 6: **The same ladder on the paired statistic instead of the count.** Each rule minus the held uniform control in mean test $R^2$ over the same 64 shared seeds, bold where the two-sided Wilcoxon signed-rank $p < 0.05$. Same runs and protocol as Table 5. Generated by `scripts/vcurve_paired.py` from `logDF_{crenel,poisson1d}_V{V}`, which is Table 5's runs without its `logET` columns.

| | | $V/B = 1.25$ | $V/B = 2.5$ | $V/B = 5$ | $V/B = 20$ | $V/B = 30$ |
|---|---|---|---|---|---|---|
| `crenel` | RAD | **+0.070** | **+0.035** | +0.006 | **−0.077** | **−0.050** |
| | PINNACLE | **+0.077** | **+0.057** | **−0.159** | **−0.320** | **−0.300** |
| | BADGE | **+0.083** | **+0.061** | **−0.093** | **−0.244** | **−0.169** |
| | GRAD-MATCH | **+0.090** | **+0.151** | **+0.171** | **+0.177** | **+0.187** |
| | ours (addpiv) | **+0.145** | **+0.156** | **+0.162** | **+0.168** | **+0.167** |
| | ours (kepiv) | **+0.142** | **+0.150** | **+0.171** | **+0.185** | **+0.189** |
| `poisson1d` | RAD | **+4.503** | **+4.306** | **+4.492** | **+3.200** | **+3.507** |
| | PINNACLE | **+4.478** | **+4.244** | **+4.488** | **+2.195** | **+3.707** |
| | BADGE | **+4.487** | **+4.303** | **+4.042** | **+4.126** | **+3.880** |
| | GRAD-MATCH | **+4.409** | **+4.607** | **+4.607** | **+4.591** | **+4.608** |
| | ours (addpiv) | **+4.592** | **+4.603** | **+3.109** | −4.311 | −4.569 |
| | ours (kepiv) | **+4.560** | **+4.564** | **+3.749** | +0.032 | −2.867 |

# F THE FULL COST GRID

Table 7: Selection FLOP ($\times 10^9$) and its share of the run's total, at $B = 40$, 4000 steps, re-selection every 100, priced for a two-input $(x, t)$ task. The upper block scores the whole pool, $V = N$; the lower one restricts each rule to a random window of $V = 200$. The training term is $4.12 \times 10^9$ for every row. At one input those rows read 29.6, 28.9, 28.9, which is what Table 1 charges. Generated by `scripts/cost_table.py` from the function the harness charges runs with, so a row cannot drift from what a run was billed. No run generates this table. The settings priced are a run's, `logDF_crenel_V200`'s, for the lower block; the upper block prices the same rules at $V = N$, which is what `logDB_{task}_s100` runs the published arms at and no run here runs ours at.

| | $N = 1{,}200$ | | $N = 20{,}000$ | | $N = 200{,}000$ | |
|---|---|---|---|---|---|---|
| | FLOP | share | FLOP | share | FLOP | share |
| RAD | 1.2 | 23% | 20.6 | 83% | 205.8 | 98% |
| PINNACLE | 30.0 | 88% | 500.0 | 99% | 5000.2 | 100% |
| GRAD-MATCH | 29.3 | 88% | 487.7 | 99% | 4876.3 | 100% |
| BADGE | 29.3 | 88% | 487.7 | 99% | 4876.3 | 100% |
| *the same rules, restricted to a random window of* $V = 200$ | | | | | | |
| RAD | 0.2 | 5% | 0.2 | 5% | 0.2 | 5% |
| PINNACLE | 5.0 | 55% | 5.0 | 55% | 5.0 | 55% |
| GRAD-MATCH | 4.9 | 54% | 4.9 | 54% | 4.9 | 54% |
| BADGE | 4.9 | 54% | 4.9 | 54% | 4.9 | 54% |
| ours | 5.3 | 56% | 5.3 | 56% | 5.3 | 56% |

**The share is a property of the budget, so here it is at a published one.** The training term scales with the budget, the network and the step count; the selection term does not, so the 88% of Section 3 cannot survive at the scale the field runs. Repricing the same analytic model at the PINNACLE benchmark's own stated setting, a $5 \times 100$ network (40,801 parameters), $B = 8192$ collocation points and 20,000 iterations (Hao et al., 2024), with $V = 200$ and re-selection every 100 as before. PINNACLE there is Hao et al.'s *benchmark suite* and is not PINNACLE, the selection rule of Lau et al. (2024) that this paper runs as an arm; the two names collide and the objects do not:

| | $N = 20{,}000$ | | $N = 200{,}000$ | |
|---|---|---|---|---|
| | FLOP ($\times 10^9$) | share | FLOP ($\times 10^9$) | share |
| RAD | 967 | 2% | 9,672 | 20% |
| PINNACLE | 22,857 | 37% | 228,573 | 85% |
| GRAD-MATCH | 30,279 | 43% | 302,787 | 88% |
| BADGE | 30,279 | 43% | 302,787 | 88% |
| *the same rules, restricted to a random window of* $V = 200$ | | | | |
| RAD | 9.7 | 0% | 9.7 | 0% |
| PINNACLE | 229 | 1% | 229 | 1% |
| GRAD-MATCH | 303 | 1% | 303 | 1% |
| BADGE | 303 | 1% | 303 | 1% |

**The headline weakens a great deal and survives only at the largest pool.** Compare the two budgets at a matched pool size. At a pool of 20,000 points our 99% becomes 37 to 43% at that benchmark's budget; at 200,000 our 100% becomes 85 to 88%. And at our own pool of 1,200 points, where we report 88%, its budget spends 3 to 4% on selection, which is not a cost worth discussing. What improves is the windowed side, because the training term it is measured against is three orders of magnitude larger: a windowed rule spends 1% of the run on selection rather than 55%. Reproduce with `cost_table.py --budget 8192 --steps 20000 --hidden 100 100 100 100 100`.

## G THE TIMING SWEEP IN FULL

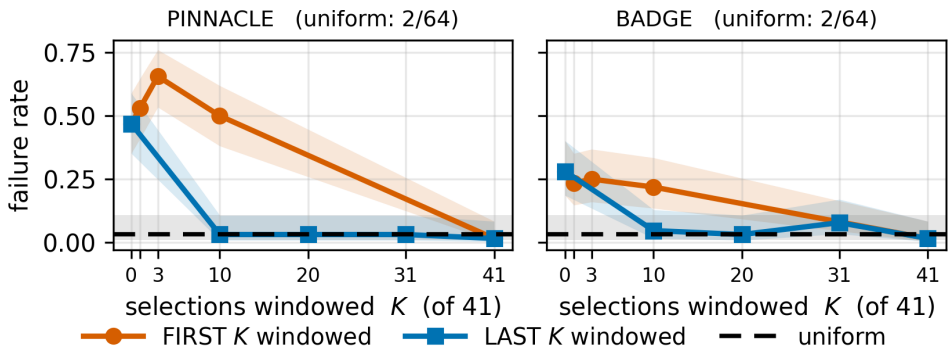



Figure 3: The same intervention, moved. Restricting a rule to $V = 100$ for $K$ of the 41 selections, with $K$ counted from the start (vermillion, circles) or from the end (blue, squares); a failure is $R^2 < 0.5$, bands are 95% Wilson intervals and the dashed black line with its grey band is the uniform control. Same task, same $V$, same 64 seeds, and the two arms were run on different $K$ grids, $\{0, 1, 3, 10, 41\}$ and $\{0, 10, 20, 31, 41\}$, so each line crosses rungs the other holds; they agree by construction at $K = 0$ and $K = 41$, which leaves $K = 10$ as the only rung both were run at where they differ. A quarter of the schedule placed at the end recovers everything; the same quarter at the start recovers nothing. Generated by `scripts/fig_timing.py` from `logDN_crenel_V100`, the first $K$ windowed, and `logDN2_crenel_V100`, the last $K$.

**Early, it may be worse than nothing, and we cannot establish that it is.** PINNACLE's failure count along the vermillion curve runs $30, 34, 42, 32, 1$ of 64 at $K = 0, 1, 3, 10, 41$. The $K = 3$ cell is *above* the unwindowed arm, by a paired mean of $-0.070$ at a two-sided Wilcoxon $p = 0.031$ over the same 64 seeds, and that $p$ does not survive correcting over the ten cells of this sweep. BADGE shows nothing of the kind $(18, 15, 16, 14, 1)$. So the figure is consistent with early windowing actively hurting one of the two arms, and one uncorrected cell is not evidence that it does.

## H MEASURED SELECTION TIME

Seconds are the `sel_seconds` column of `runs/logDF_{crenel,poisson1d}_V*`, the same runs the $V$ ladder reads. Windowing compares a rule against itself, so implementation constants cancel, which is what a between-method cost claim cannot offer. Over the same $64$ seeds, GRAD-MATCH's selection time per run is $0.85$, $1.87$, $2.84$, $12.03$ and $18.54$ seconds at $V = 50, 100, 200, 800, 1200$ on `poisson1d`: linear in $V$, which is the $O(V)$ claim measured rather than restated. The $V = 1200 \to 200$ ratio is $6.52\times$ there and $4.75\times$ on `crenel`, against the $6.0\times$ the FLOP model of Table 7 predicts. The model is an order of magnitude, not a timing: it counts arithmetic and the seconds include Python, and on a $4{,}353$-parameter network the fixed cost per selection is a visible share of the smaller cells. Both figures come from the `logDF_{task}_V{V}` stamps, as the ratios above do; `logDB` records no `sel_seconds` column at all, so no timing here can come from the runs Table 1 reads.

§B says our selector's many small Nyström factorisations vectorise far worse than RAD's single vectorised residual pass, so its wall clock is worse relative to RAD than its FLOP count implies. Here it is. Both windowed at $V = 200$, same $64$ seeds, same runs: ours takes $0.974$ seconds against RAD's $0.017$ on `crenel` and $2.744$ against $0.028$ on `poisson1d`, which is $57\times$ and $98\times$ where the FLOP model of Table 7 says $26\times$. The disagreement is a factor of two to four, which is why every cost claim the paper leans on is within-rule: the same code path, windowed against not.

## I WHAT ELSE CHANGES WITH $V$: THE REFERENCE SET, THE ESTIMATOR, THE BOUNDARY WEIGHT AND THE QUANTILE

**PINNACLE's reference set is fixed at $64$ points except at the smallest rung.** Its score is a Nyström self-energy $\hat{\alpha}(z) = r(z)^2 k_z^\top K_{\mathrm{ref}}^{-1} k_z$ over a reference set of $n_{\mathrm{ref}} = \min(64, V)$ candidates, truncated to its top 16 eigenpairs (Lau et al., 2024). The ladder is $V \in \{50, 100, 200, 800, 1200\}$, so $n_{\mathrm{ref}}$ is $50, 64, 64, 64, 64$: **the reference *count* changes once, between the first rung and the second, and is the same at the other four**. The count is not the whole of the estimator. The reference points are drawn uniformly from the window, so at $V = 50$ the reference set *is* the candidate set and the self-energy is exact, while at $V = 1200$ the same $64$ points stand in for a window nineteen times larger. What holds across four rungs is the count; the fraction falls from $64\%$ to $5.3\%$, and the paragraph after next is about that.

The measured ladder puts the crossing where the estimator is constant. PINNACLE's failures on `crenel` are $0, 1, 18, 27, 30$; the change across the estimator boundary is $0 \to 1$ and the movement is $1 \to 30$ across four rungs sharing one estimator. The crossing at $V/B = 5$ sits between the second and third. So the confound in the count is real, it is confined to the leftmost point, and it is not what the curve is made of. The confound in the fraction is confined nowhere, and the two are not one thing. What we cannot rule out from these runs is a milder version: that a reference set covering $64\%$ of the candidates behaves differently from one covering $5\%$ even at fixed size. Pinning $n_{\mathrm{ref}}$ below $50$ for every rung would settle it and is not run.

**The kernel estimator is not the same object at every rung either.** PINNACLE's reference set is the case above; the same exposure reaches BADGE and GRAD-MATCH by another route. Both build their embedding from a rank-$\leq 64$ Nyström factor of the window's tangent features, with the landmarks drawn from the window itself, so the fraction of the candidate set serving as landmarks runs $100\%$, $64\%$, $32\%$, $8\%$, $5.3\%$ across the ladder. At $V = 50$ the factor is exact; at $V = 1200$ it is a rank-$64$ sketch of a $1200$-point kernel.

**It runs in opposite directions for the two things it touches.** For PINNACLE and BADGE, which degrade in $V$, a criterion estimated worse at every rung is an alternative account of that degradation, and these runs cannot separate it from ours. For GRAD-MATCH, which improves in $V$, the sketch's *quality* runs the other way, the one rule the dividing line calls safe getting better while its own estimator gets worse; its matching *problem* does not, which the next paragraph measures. Pinning the landmark count below $50$ at every rung would settle the first and is not run.

**The sharpest version of this objection is not closed.** A fixed-rank sketch against a growing dictionary might make GRAD-MATCH's matching problem easier on its own, which would supply the

improving half with no mechanism in it. *Relative match error* here is $\|\rho\|/\|t\|$, the norm of what the pursuit still has to match after its $B$ picks, over the norm of the target $t = \sum_j r_j R_j$ it started from. `scripts/match_error.py` reproduces `nystrom_factor` and `gradmatch_select`'s OMP loop on a power-law feature spectrum with six draws a rung, and the error *falls* across the ladder: from 0.028–0.065 at $V = 50$ to 0.018–0.025 at $V = 1{,}200$, in all four variants we ran (two spectral decays crossed with drawing each rung's pool afresh or nesting it inside the largest, `--decay {1,2} --draw {nested,independent}`, each saved as its own run). It worsens where the rank cap first binds, at the rung where the kept rank reaches $m = 64$, and recovers after.

So a wider candidate set does make the matching problem easier on a surrogate spectrum, and the improving half of GRAD-MATCH's behaviour has an alternative account that these runs do not exclude. The features are not ours and this bounds nothing either way; what it removes is our claim to have ruled the objection out.

**"The boundary weight is a function of the selected set, so $V$ moves the objective too."** It would, and it is switched off. The harness can train $\overline{r^2} + \lambda_{\mathrm{bc}}\,\overline{b^2}$ with $\lambda_{\mathrm{bc}} = \mathrm{Tr}(K_{rr})/\mathrm{Tr}(K_{uu})$ recomputed from the selected points, which would make the objective itself a function of $S$ and the flow $\dot{e} = -JWJ^\top e$ rather than $-JJ^\top e$. It is off by default and it is off in every run this paper reports: `trace_balance` is `false` in `config.yaml` and `False` in the metadata of every run behind every table here, so $\lambda_{\mathrm{bc}} = 1$ and the objective is identical across arms.

**What is graded is the solution and what is measured is the residual.** Section 5's dissociation puts a one-step *residual* decrease, $e^\top K[:, S]e_S$, against a trained *solution* error, $R^2(u_\theta, u^\star)$ on a dense grid.

It is not a stability artefact on the task where we measure it. `poisson1d` is a well-posed linear elliptic problem with Dirichlet data, so a small $L^2$ residual bounds the solution error, and $e^\top K[:, S]e_S$ is written for the decrease of the *full-domain* error rather than the error at the selected points, so it is not the concentration story in disguise either.

But we never report the residual norm. Runs store $R^2$ and its mean squared error against $u^\star$ and nothing else, so a reader cannot check that the two quantities moved apart rather than one of them being the wrong instrument. That is a reporting gap and not an argument, and what it costs to close is a re-run rather than a script: the harness saves no model weights, so the residual at the end of a run cannot be recovered from anything on disk. One column in `per_seed.csv` and one sweep would do it.

The boundary budget, since it is standard to report it here and we did not: the 1-D problems carry exactly two Dirichlet points, at $x = 0$ and $x = 1$, outside the selected budget $B$ and never selected against; the space–time problems carry $n_{\mathrm{bc}} = 80$ or 90 split evenly across edges. Only the interior collocation set is under selection anywhere in this paper.

**RAD samples rather than taking a top-$B$, so the identity reaches it through the tail.** The quantile identity of Section 2 is stated for a rule that keeps the best $B$ of $V$. RAD does not: it draws $B$ points from $p \propto \varepsilon^k/\overline{\varepsilon^k} + c$ over the candidate set (Wu et al., 2023).

None of this is a derivation. It is the reading under which RAD belongs on the same axis as the others, and it makes a prediction we have not tested: a rule that samples with a $V$-independent floor should be flat in $V$ once $c$ is large enough, and the $c$ sweep of Appendix U runs at two values rather than a ladder.

## J THE CROSSING UNDER EVERY ANALYSIS WE COULD HAVE CHOSEN

So here is the headline under the whole grid at once.

**These thirty are not pre-registered, which changes what they are worth.** The grid was specified after the runs it re-analyses, there is no design note for it, and `scripts/specification.py` was written in the same cycle as this appendix. What *is* pre-registered is the failure threshold, the two-sided admissibility rule and the multiplicity families, each committed before the runs they judge (Appendix A); the grid that varies them is not. The abstract called these choices pre-registered until

we corrected it; the word is gone from the paper. A robustness grid chosen afterwards answers the weaker of two questions. It can show that a result is not an artefact of the four choices we thought to vary. It cannot show that we stopped varying them for any reason other than that the answer had stopped changing, and no reader can check that from the outside.

**The forks.** All are re-analyses of the runs already described, and none needs a run. The failure threshold ($R^2 < 0.3, 0.5, 0.7, 0.9$); the statistic (failure count with Fisher, or the paired mean with Wilcoxon, which the threshold does not touch); the correction (raw, or Holm over that panel's thirty tests); and the seed set (all 64, the first 32, the last 32). That is thirty specifications. A rule *crosses* if at some rung it is significantly worse than the uniform control.

Table 8: **Thirty analysis choices, and how many of them each rule crosses under.** The forks are the failure threshold ($R^2 < 0.3, 0.5, 0.7, 0.9$), the statistic, the multiplicity correction and the seed set; a rule *crosses* if at some rung it is significantly worse than the uniform control. All thirty are re-analyses of the runs behind Table 5 and none needs a run. Generated by `scripts/specification.py`.

| rule | specifications in which it crosses | share |
|---|---|---|
| RAD | $8/30$ | $27\%$ |
| PINNACLE | $24/30$ | $80\%$ |
| BADGE | $19/30$ | $63\%$ |
| GRAD-MATCH | $0/30$ | $0\%$ |

"Three of four published rules cross" holds in 8 of the 30. Two rules cross in most specifications and RAD in about a quarter of them. The claim in this paper is now the specification-robust one: **two published rules cross under most defensible analyses and a third under a minority**.

At $B = 40$, GRAD-MATCH crosses in **none** of the thirty. The one claim this paper makes about a set-level objective is the only one of the four that no analysis choice can overturn.

**Where the crossing disappears entirely.** Six specifications have no rule crossing, and all six sit at a threshold end: $R^2 < 0.9$ with a correction or a half of the seeds, where the control itself fails $56/64$ and the cell is near-void by Section 2's own rule; and $R^2 < 0.3$ under Holm on half the seeds, where almost nothing fails and there is little to detect. Appendix Q reaches the same two ends by sweeping the threshold alone. The crossing is a mid-threshold, whole-seed phenomenon and this says so in one table.

**Across all five tasks, a hundred and fifty specifications.** Running the same grid on the four tasks of Appendix K, PINNACLE crosses in 2 of 30 on `poisson1d` and RAD in 2 of 30 on `staircase`, and nothing crosses anywhere else. All four of those cells are the paired mean on the *first half* of the seeds, and each vanishes on the full set and on the other half. So the reversal being one task is not a property of the analysis we happened to choose: it is one task across a hundred and fifty specifications. And GRAD-MATCH crosses in 0 of 150, all of them at $B = 40$; Appendix X is where that stops holding.

**The mechanism measurement has a fork too, and it is the worse finding.** Section 5's "the rules are not failing to optimise" is read at one checkpoint. The same runs record several. Percentage change in the one-step decrease from $V/B = 1.25$ to 30, mean over the eight seeds of each seed's median, which reproduces the two figures the section quotes, from `runs/logDM2_{task}` by `scripts/checkpoints.py`:

Table 9: **The mechanism measurement read at every recorded checkpoint, not one.** Percentage change in the one-step loss decrease from $V/B = 1.25$ to 30, mean over the eight seeds of each seed's median. Bold marks the cells the text quotes, not significance. Generated by `scripts/checkpoints.py` from `runs/logDM2_{task}`.

| step | 0 | 25 | 50 | 100 | 200 | 500 |
|---|---|---|---|---|---|---|
| `poisson1d`, PINNACLE | $-95\%$ | $+129\%$ | $\mathbf{+405\%}$ | – | – | – |
| `poisson1d`, GRAD-MATCH | $-45\%$ | $-22\%$ | $+73\%$ | – | – | – |
| `crenel`, PINNACLE | $+31\%$ | $+23\%$ | $\mathbf{-40\%}$ | $-4\%$ | $+30\%$ | $\mathbf{+619\%}$ |
| `crenel`, GRAD-MATCH | $+1\%$ | $-26\%$ | $\mathbf{-40\%}$ | $+14\%$ | $+112\%$ | $+242\%$ |

In 7 of the 18 cells printed above, widening $V$ buys *less* of the objective, not more, and in 14 of the 36 the same sweep holds over all four rules including the two not shown. The $+405\%$ and $+73\%$ the section quotes are `poisson1d` at step 50. **At the same step on `crenel` the figure is $-40\%$**, and across `crenel`'s six checkpoints it is not monotone in any direction. We report the initialisation reversal and this one.

So the mechanism is demonstrated on `poisson1d` and not on `crenel`, where widening $V$ buys *less* of the one-step decrease at the same step. Against a control re-drawn on the rules' own schedule seven of this panel's eight rule arms cross there (Section 4), so the account and the phenomenon are now on the same task and the same rule: on `poisson1d` PINNACLE buys $405\%$ more of the proxy, ends worse, and crosses below the control at $V/B = 30$, uncorrected. What does not survive is any claim that the account is demonstrated on `crenel`: there the one-step figure runs the other way and we have no account of that.

**Why a multiverse at all, when the predictions were registered.** Because the two answer different objections. Gelman & Loken (2013) argue that "researcher degrees of freedom can lead to a multiple comparisons problem, even in settings where researchers perform only a single analysis on their data", the number of comparisons being the ones that *would* have been made had the data come out otherwise. A registration fixes the analysis we ran; it does not show that the analyses we would have run instead say the same thing. The grid does, for the four axes it covers.

**And one team's grid is a lower bound on the variation.** Silberzahn et al. (2018) gave one data set and one question to "29 teams that analyzed the same data set to answer the same research question", and found that neither the analysts' prior beliefs nor their expertise nor peer ratings of analysis quality "readily explained the variation in the outcomes". Our thirty specifications are four axes one team thought of. A different team would draw different axes, and nothing here bounds that.

**The idea is not ours, and it has a name we should use correctly.** Steegen et al. (2016) propose "performing all analyses across the whole set of alternatively processed data sets corresponding to a large set of reasonable scenarios", which is the construct above, though their axes are data-processing choices and ours are choices of statistic, threshold, correction and seed subset. Simonsohn et al. (2020)'s specification curve is a different object: three steps, of which the third is "conducting joint inference across all specifications". We do not do that third step, so this is a multiverse and not a specification curve, and we call it one.

In machine learning the idea is Bell et al. (2022)'s, who state their barrier plainly: "the size of the search space and the presence of continuous dimensions ... makes exhaustive search intractable", and model the multiverse with a Gaussian-process surrogate instead. Our grid is exhaustive only because it is small and discrete: four thresholds, two statistics, two corrections, three seed sets, five tasks, every cell computed. That is a property of our axes, not an improvement on their method.

The closest published form of what we do is Simson et al. (2024)'s, who do enumerate: "by combining decisions, we create a grid of all possible 'universes' of decision combinations", and who deliberately separate *evaluation* decisions from the data-processing ones Steegen treats. Their motivation is ours reversed onto a different quantity, that a chosen path can be selected to flatter a metric, and every axis of our grid is an evaluation decision in their sense. We differ in the object, a selection rule's failure rate rather than a fairness score, and in nothing about the method. Generated by `scripts/specification.py`.

## K THE SAME LADDER ON FIVE TASKS

`logDF_{crenel,poisson1d}` and `logEH_{sinjump,staircase,tri_zone}` are one experiment. All twenty-five run directories carry the same hidden widths, pool size, activation, step count, re-selection cadence, budget and seed count: $[64, 64]$, $N = 1{,}200$, $\tanh$, $4000$ steps, $c = 100$, $B = 40$ and $64$ seeds, one signature across all twenty-five. Section 4 reported two of the five and Appendix Z reported the other three as a failed replication, which they are: no rule crosses below uniform on any of them under the registered analysis; Appendix J finds RAD crossing on `staircase` in 2 of its 30 specifications, both on half the seeds.

**What does replicate on all five is the derivative.** Each rule against itself from the smallest candidate set to the whole pool, median paired change in test $R^2$ over the shared seeds, bold where it survives Holm over the whole grid of thirty:

Table 10: **Each rule against itself, on all five ladder tasks.** Median paired change in test $R^2$ from the smallest candidate set to the whole pool over the shared seeds, bold where it survives Holm over the whole grid of thirty. A within-rule comparison needs no control. Generated by `scripts/separation.py`.

| task | RAD | PINNACLE | BADGE | GRAD-MATCH | ours (add) | ours (kepiv) |
|---|---|---|---|---|---|---|
| `crenel` | $\mathbf{-0.0926}$ | $\mathbf{-0.3290}$ | $\mathbf{-0.2313}$ | $\mathbf{+0.0690}$ | $\mathbf{+0.0209}$ | $\mathbf{+0.0371}$ |
| `poisson1d` | $-0.0025$ | $\mathbf{-0.0060}$ | $-0.0021$ | $\mathbf{+0.0035}$ | $\mathbf{-0.1735}$ | $\mathbf{-0.0236}$ |
| `sinjump` | $+0.0000$ | $\mathbf{-0.0273}$ | $-0.0085$ | $\mathbf{+0.0298}$ | $+0.0028$ | $\mathbf{+0.0030}$ |
| `staircase` | $+0.0001$ | $\mathbf{+0.0010}$ | $\mathbf{+0.0008}$ | $\mathbf{+0.0007}$ | $\mathbf{+0.0003}$ | $\mathbf{+0.0002}$ |
| `tri_zone` | $\mathbf{-0.0107}$ | $\mathbf{-0.0057}$ | $-0.0012$ | $\mathbf{+0.0035}$ | $+0.0011$ | $\mathbf{+0.0021}$ |

**The set-level rule improves on five of five and every one survives Holm. The *dichotomy* holds on four of the five.** Five of five is the count on this ladder and not in the paper: on `burgers1d` at $\nu = 0.01/\pi$ (Appendix M) the set-level rule is flat rather than improving, $+0.016$ at $p = 0.30$, while the other six arms there degrade. On `staircase` it does not hold at all: PINNACLE, BADGE and GRAD-MATCH all improve and all survive Holm, and the set-level rule improves *least* of the three, $+0.00072$ against $+0.00095$ and $+0.00081$, which the table above rounds to $+0.0007$, $+0.0010$ and $+0.0008$. That cell is saturated in the sense of Section 2's two-sided void rule, a control that cannot be lost to, which is not the sense Section 3 uses for `convdiff1d`, where the word means no headroom left to win, which disqualifies its two pointwise entries and keeps its set-level one toward "five of five", which is the same cell counted two ways.

**Our own two arms are the ablation, and they say the subtraction is not what buys it.** In both, selecting a point changes every other point's score, which is the property Section 5's test asks for. So the pair does not isolate one mechanism from another; it shows that satisfying the test's stated property is not enough.

The two columns are not intermediate between the other four. They *improve* on `crenel`, where the three pointwise rules degrade worst, and they degrade on `poisson1d` by $-0.1735$ and $-0.0236$. At the paper's own failure threshold that task separates one arm from all five others: GRAD-MATCH goes $2/64$ to $0/64$ while RAD, PINNACLE and BADGE go $4 \to 10$, $2 \to 11$ and $2 \to 10$, and ours $1 \to 24$ and $2 \to 26$.

**What this is not.** It is not a one-ingredient ablation, and three differences other than the objective could carry the `poisson1d` cell. Our arms keep $b = B/2$ per round, so their ladder is $V/b = 2.5$ to 60 where every baseline's is $V/B = 1.25$ to 30; they retain half the budget across exams where the baselines rebuild; and the `add` score divides both of its halves by a window extremum, which is itself a statistic of $V$. Each is an account of that cell that has nothing to do with set objectives, and we have not separated them.

So the dichotomy is not *whether the selection step subtracts*, which is the natural reading of Section 5's test and the first repair a reader will propose; a rule can change every other point's score at every pick and still collapse on a task where the set-level rule does not. What survives is the narrower statement that the objective must be a function of the selected set. Our own method is the counterexample, and so, by Appendix AA, are PINNACLE and BADGE, which carry the same route and cross below uniform anyway; the cheap repair is ruled out without us. The headroom criterion is ours and post hoc besides, the one Appendix Z applied to drop the task after the run. Read straight: four tasks separate the two kinds of rule, and on the fifth nothing does.

**And the five are not five independent tasks.** `crenel`, `sinjump`, `staircase` and `tri_zone` are all one-dimensional scalar fits on the same interval through the same $2 \times 64$ network, differing in how many jumps they carry and how sharp; `src/data.py` records two of them as ported from the same earlier study. Only `poisson1d` is a PDE. They share seeds 0–63 and their initialisation stream, so Holm over the grid of thirty corrects across tests that are not exchangeable, and "five of five" is four variants of one generator family plus one PDE rather than five draws from the space of tasks.

The magnitudes are not equal and the count hides that: `crenel` moves PINNACLE by $-0.329$ where `tri_zone` moves it by $-0.006$ and `staircase` by $+0.001$.

**What is not here.** Pairing the four published rules against each other, 2 of 30 task-by-pair contests resolve at $V/B = 1.25$ against 19 of 30 at the whole pool, which reads as a claim that a benchmark cannot tell these methods apart unless $V/B$ is pinned. The intermediate rungs refuse it: 18, 20, 18, 19 of 30 at $V/B = 2.5, 5, 20, 30$, or 1, 14, 18, 17, 17 after Holm over the thirty, which is the correction this paper reports elsewhere and which does not change the shape. The entire step is between the first two rungs, and at $V/B = 1.25$ a rule keeps forty of fifty uniform draws and can move at most ten points, so most of the agreement there is arithmetic. Every setting the field runs sits above the step. Five of the thirty contests are PINNACLE against BADGE, whose per-point scores Appendix P shows are one quantity; their selected sets still differ, but a sixth of this grid is two labels on one score. Generated by `scripts/separation.py --ours --tex-within`; the script's default flags print a different table, which this paper does not use.

## L THE RULE THE IDENTITY IS EXACT FOR, ON THE AXIS

Every proposition in Appendix C about truncation is stated for a rule that keeps the best $B$ of $V$, and the framing sentence of this paper's abstract is exact only for such a rule. **No *softened* arm in Section 3 or Section 4 is one**, and the two that are, RAR-G and dynamicAL, enter in Section 4 and are the subject of this appendix. RAD draws from a density $p \propto \epsilon^k/\overline{\epsilon^k} + c$, so the quantile reaches it through the tail of $\epsilon$; PINNACLE and BADGE take one $\arg\max$ and then $k$-means++, so only the first of forty picks is a best-of-$V$; GRAD-MATCH is orthogonal matching pursuit.

**And it is not one rule.** dynamicAL (Wang et al., 2022a) selects by $\mathrm{argsort}(-d)[:B]$ exactly as RAR-G does, but $d$ is the training-dynamics score on the eNTK factor, $d_u = r_u^2\|R_u\|^2 + \gamma\, r_u(RR^\top r)_u$, which shares nothing with a residual magnitude but the residual itself. It is level with the re-drawn control at $V/B = 1.25$ on both tasks ($p = 0.55$ and $0.18$), crosses at 2.5 on both, and fails every seed of both from $V/B = 20$; within-rule it moves $-1.688$ and $-4.548$, at $p = 3.5 \times 10^{-12}$ and $1.6 \times 10^{-10}$. **Two scores with nothing in common, collapsing at the same rung on both tasks, is the selection form and not the field it scores.** That was the one reading the RAR-G ladder alone could not rule out, and it is what P-EU2 was registered to settle.

**The multiplicity family, stated once and used on both panels of this ladder.** Every rule arm on a ladder against the re-drawn control at all five rungs by two-sided Fisher: nine arms and forty-five tests on `crenel`, eight and forty on `poisson1d`, Holm over them. The two differ because `RAD+cov` runs on the `crenel` ladder at all five rungs (Table 16) and on no rung of the other, and an earlier statement of this family counted eight on both. The correction changes no verdict and every quoted value slightly: on `crenel`, Holm over forty-five gives the two top-$B$ rules $3.8 \times 10^{-36}$ where forty gave $3.3 \times 10^{-36}$, PINNACLE $6.5 \times 10^{-5}$ where it gave $5.5 \times 10^{-5}$, BADGE $0.041$ where it gave $0.035$, and RAD $0.020$ where it gave $0.017$. The numbers printed below are the forty-test ones, which is the family this appendix was computed under, and they are all that changes. On `crenel` five arms cross and the order is the softening order, the two top-$B$ rules at $V/B = 2.5$ (adjusted $3.3 \times 10^{-36}$ each), PINNACLE and BADGE at 5 ($5.5 \times 10^{-5}$, $0.035$) and RAD at 20 ($0.017$), with GRAD-MATCH and both of ours never. On `poisson1d` four cross and the order is *not* the softening order: the two top-$B$ rules at 2.5 ($9.6 \times 10^{-6}$, $4.2 \times 10^{-6}$) and both of ours at 20 ($7.1 \times 10^{-7}$, $0.0031$), with all four published softened rules never.

**It is one line of configuration, it was not our rule to invent, and it is not exactly theirs either.** RAR-G is `add_anchors` of the $m$ largest residuals, the reference implementation quoted in Section 1 (Lu et al., 2021), and Wu et al. (2023) run it as a baseline under that name. In our harness it is the RAD selector with the sampling step replaced by the top $B$ of $\epsilon$; $k$ and $c$ cannot change that ordering and are pinned so the arm is unambiguous. **One departure matters, it is not a detail, and we ran it.** As published, RAR-G *adds* its $m$ points to a training set that keeps everything it already had; the arm above *replaces* a set of fixed size $B$ from a fresh window at every exam, because a fixed budget is what every other arm in this paper runs at and what makes them comparable. Before running it we registered the conjecture that the growing set keeps the uniform coverage it started with, and that this is why the shipped version does not collapse. **The conjecture was load-bearing and it is confirmed.**

**And the same departure applies to dynamicAL, which we said of RAR-G and did not say of it.** Wang et al. (2022a) set their method in pool-based active learning, in which "there are $R$ rounds in total, and for each round, a query set $Q$ consisting of $b$ unlabeled samples can be queried", for a total budget of $b$ times the number of rounds. The queried points join the labelled set, so their budget *accumulates* across rounds exactly as `add_anchors` does. Our dynamicAL arm replaces a set of fixed size $B$ at every exam, for the same reason the RAR-G arm does, and that is a second departure from a published method rather than one. It is the same departure, so the conjecture confirmed above covers it; we have not run a growing dynamicAL and do not claim the growing version collapses.

**And BADGE's budget regime is ours as well.** Ash et al. (2020)'s Algorithm 1 draws an initial labelled set $S$, then at each of $T$ iterations scores the unlabelled pool minus $S$, queries a batch of $B$ labels, and then "retrains the model, and repeats". The candidate pool excludes what is already labelled and $S$ grows by $B$ a round, so its budget accumulates to $M + TB$. Our BADGE arm replaces a set of fixed size at every exam, as the RAR-G and dynamicAL arms do and for the same reason, and that is a third departure of the same kind rather than a third kind of departure. The conjecture confirmed above covers all three; we have run none of them in its growing form.

**What this costs the paper, said plainly.** On `poisson1d` the four arms that cross under correction are these two top-$B$ rules and our own two. Both top-$B$ arms are ours at a fixed budget and neither published rule ships that way, so on that task *no published rule, run in its own budget regime, crosses*. The four published softened rules never cross there, and the crossing we report on `poisson1d` is a statement about a budget regime we imposed. The `crenel` ladder is where published softened rules cross on their own terms, PINNACLE and BADGE at $V/B = 5$ and RAD at 20, and the Burgers ladder of Appendix M is where that repeats.

`rar_grow_exam` starts from the same 40 uniform points and appends the largest residual in a window of $V$ at each of the forty exams, ending at 80 points, which is `add_anchors` semantics. `random_grow_exam` appends a uniform point instead, because a growing set differs from a fixed one in two ways, the growth and the rule that picks what is added, and without the second arm the comparison confounds them. Over the same ladder and the same 64 seeds:

| $V/B$ | 1.25 | 2.5 | 5 | 20 | 30 |
|---|---|---|---|---|---|
| `crenel`, grow by residual | 0.983 | 0.984 | 0.986 | 0.985 | 0.986 |
| `crenel`, grow by uniform | 0.894 | 0.899 | 0.901 | 0.902 | 0.890 |
| `poisson1d`, grow by residual | 0.980 | 0.983 | 0.968 | 0.996 | 0.989 |
| `poisson1d`, grow by uniform | 0.583 | 0.800 | 0.915 | 0.718 | 0.939 |

It fails 0 of 64 at every rung of `crenel` and 0 to 2 on `poisson1d`. Its within-rule change across the ladder is $+0.003$ ($p = 0.041$) and $+0.010$ ($p = 0.18$), so **P-EV1 holds**: no degradation in $V$, against the same rule's $-1.76$ and $-7.38$ at a fixed budget. **P-EV2 holds** too, and more strongly than registered: appending by residual beats appending uniformly by $+0.05$ to $+0.40$ at every rung of both tasks, all at $p \leq 0.0024$. *Searching harder helps here.*

**And the arm that takes no harm is the one searching further into the tail.** The two curves of Figure 1 are drawn against the same $V$, but they do not keep the same share of it. The fixed-budget arm keeps the best $B = 40$ of $V$, which is the $1 - B/V$ quantile: 0.20 at $V = 50$ and 0.967 at $V = 1{,}200$. The growing arm keeps the best *one* of the same window, which is $1 - 1/V$: 0.98 and 0.99917 at the same two rungs, a kept fraction 40 times smaller at every rung. So the arm that collapses is selecting at a *gentler* quantile than the arm that does not, at all five rungs, and the paper's conclusion is sharper than the figure's $x$ axis can show: the harm is searching hard *at a fixed budget*, and not searching hard.

**The cost of saying that is a confound, and it is ours to name.** A growing arm differs from a fixed one in the budget *and* in how many points it takes per round, and the two move together here because taking one point per round is what `add_anchors` does. The held-at-$B = 80$ arm above removes the budget difference and leaves the round size, and it collapses, so the budget is carrying weight; no arm here removes the round size at a fixed budget, and we do not claim the decomposition. What the pair does establish is the direction: the quantile alone does not predict the sign, because the extreme of the two quantiles here is the safe one.

**The comparison Wu et al. (2023) make is not the one above, and we owe it.** They set their growing rule against uniform at the *same final count*, and report "RAR-G and RAR-D are even worse than the Random sampling" at a very small number of residual points. Ours ends at 80, so the matched control is a uniform draw of 80: `logEO_{task}_B80`, the budget ladder's own cell at the same pool and network. On the 32 seeds the two runs share, and unpaired because they are two runs, the growing arm is *above* both uniform controls at 80 points on both tasks: $+0.059$ over a held draw and $+0.063$ over a re-drawn one on `crenel` (Mann–Whitney $p \approx 10^{-11}$), and $+0.306$ over held ($p = 9 \times 10^{-6}$) and $+0.014$ over re-drawn ($p = 0.048$) on `poisson1d`. Those two `poisson1d` figures are the first rung's. The margin over the re-drawn control barely moves across the ladder, $+0.014$, $+0.012$, $+0.009$, $+0.013$, $+0.010$, and its $p$ runs $0.048$, $0.084$, $0.157$, $0.027$, $0.311$, so it reaches $0.05$ at two rungs of five and the one we quoted is among them. The margin over the held control is $+0.30$ at every rung and significant at every rung. We name the rung because the re-drawn comparison is the one whose verdict depends on it. This does not contradict Wu et al.: their regime is "a very small number of residual points", and our own budget sweep puts 80 past the point where failure stops moving.

**The end budget is not what saves it, and that was registered before it was run.** The growing arm finishes at 80 points while every fixed arm holds 40, so "growth protects" and "eighty points protect" were not separated by anything on disk. P-EW1 and P-EW2 (`notes/fixed_budget_grow_control.md`, frozen with a refutation branch) asked for the same rule held at a fixed $B = 80$, and it was run at $N = 20{,}000$ on 32 shared seeds with $V/B \in \{1.25, 2.5, 10, 30\}$. **Both hold.** At $V/B = 1.25$ the fixed arm is level with both controls ($0.911$ against a held $0.905$ and a re-drawn $0.896$ on `crenel`; $0.978$ against $0.794$ and $0.993$ on `poisson1d`), which is the left endpoint Proposition 3 fixes; by $V/B = 30$ it fails 32 of 32 seeds on *both* tasks, a within-rule $-1.78$ and $-4.16$ at $p = 5 \times 10^{-10}$. Against the growing arm at matched rungs it is lower everywhere, by $-0.07$ to $-4.17$, significant at every cell but `poisson1d` at $V/B = 1.25$ where both sit at the ceiling and nothing separates. So the protection is the growth. The first attempt at this ladder was void because the arm was run without the windowing prefix and returned the same number at every rung; that is recorded in the note rather than deleted.

**P-EV3, and it is the cost.** The collapse this appendix reports belongs to top-$B$ selection *at a fixed budget* and not to the procedure this paper's title and opening snippet name. What that does not cost: RAD, PINNACLE and BADGE replace a fixed budget in their own papers and not only in our harness, so every crossing reported for them stands.

A growing set is not free. Its cost grows without bound, which is the cost R3 was built to remove (Daw et al., 2023), and this ladder does not price it: the FLOP model here is a lower bound for these two arms because $|S|$ changes during the run, so they are not on the frontier of Appendix N. Same protocol as every other ladder: $B = 40$, $N = 1{,}200$, $[64, 64]$ with $\tanh$, 4000 Adam steps, re-selection every 100, 64 shared seeds. RAR-G's rungs are `logET_{task}_V{V}` and dynamicAL's `logEU_{task}_V{V}`; the table is `scripts/vcurve.py --prefix logET logEU`.

Table 11: **The two top-$B$ rules on the same ladder.** Seeds failing at $R^2 < 0.5$ of 64, same protocol as Table 5: $B = 40$, $N = 1{,}200$, $[64, 64]$ with $\tanh$, 4000 Adam steps, re-selection every 100. The FLOP model is a lower bound for a growing set, so these two are not on Appendix N's frontier. Generated by `scripts/vcurve.py --prefix logET logEU` from `log{ET,EU}_{task}_V{V}`.

| $V$ | $V/B$ | uniform | re-drawn | RAR-G (top-$B$) | dynamicAL (top-$B$) |
|---|---|---|---|---|---|
| `crenel` | | | | | |
| 50 | 1.25 | 2/64 | 0/64 | 0/64 | 1/64 |
| 100 | 2.5 | 2/64 | 0/64 | 64/64 | 64/64 |
| 200 | 5 | 2/64 | 0/64 | 64/64 | 64/64 |
| 800 | 20 | 2/64 | 0/64 | 64/64 | 64/64 |
| 1200 | 30 | 2/64 | 0/64 | 64/64 | 64/64 |
| `poisson1d` | | | | | |
| 50 | 1.25 | 35/64 | 1/64 | 1/64 | 5/64 |
| 100 | 2.5 | 35/64 | 1/64 | 23/64 | 24/64 |
| 200 | 5 | 35/64 | 1/64 | 58/64 | 54/64 |
| 800 | 20 | 35/64 | 1/64 | 64/64 | 64/64 |
| 1200 | 30 | 35/64 | 1/64 | 64/64 | 64/64 |

**What it does.** At $V/B = 1.25$, where it keeps 40 of 50 and can move at most ten points, it is indistinguishable from the re-drawn control on both tasks: $+0.027$ at $p = 0.30$ on `crenel` and $+0.092$ at $p = 0.36$ on `poisson1d`. One rung later it fails 64 of 64 and 23 of 64, at Fisher $8.4 \times 10^{-38}$ and $3.2 \times 10^{-7}$ against controls at $0/64$ and $1/64$. From $V/B = 20$ it fails every seed on both. Within-rule from the first rung to the last the paired mean is $-1.758$ on `crenel` and $-7.375$ on `poisson1d`, both at $p = 3.5 \times 10^{-12}$, against $-0.377$ for the worst *softened* `win:` arm on the `crenel` ladder and $-0.996$ for the worst on `poisson1d`. **It is the sharpest sign reversal in this paper on `crenel`**, where no softened arm moves by more than a fifth of it; the other top-$B$ rule does, dynamicAL's $-1.688$ being 96% of it, which is the point of grouping them; on `poisson1d` our own `addpivoted` arm degrades further still, by $-9.16$.

**What it was registered to do.** The predictions were frozen before the 64-seed ladder and after a four-seed smoke at one rung whose result the note discloses. P-ET1 asked for a degradation larger than any `win:` arm's; P-ET2 for a crossing no later than PINNACLE's $V/B = 5$; P-ET3 for degradation on `poisson1d`, with a crossing there registered as tested and not promised. All three hold, the crossing is at $V/B = 2.5$ on both tasks, and the `poisson1d` crossing that was not promised happened.

**What it costs this paper, and it is a claim of ours it retires.** Section 4 used to report `poisson1d` as a magnitude effect on the grounds that every published rule helps at every $V$ there and none crosses. RAR-G is published, is the rule in our own opening, and crosses there at the fixed budget we impose on it. So that split was never a property of the task, and it survives only as a statement about the *held* control, which fails 35 of 64 on that panel and can hardly be crossed by anything.

**What it does not show.** That the other arms' positions are explained by their distance from a literal $\arg\max$. That ordering holds on `crenel` and fails on `poisson1d`, where the softened rules do not cross under correction at all while both of ours do. Two endpoints are added here; the ladder is not thereby a controlled comparison of selection sharpness, and two scores are not a family.

## M THE SAME LADDER ON THE CANONICAL BURGERS INSTANCE, AT A BUDGET THAT DOES NOT SOLVE IT

`burgers1d` is Burgers' equation as Raissi et al. (2019) pose it and as Wu et al. (2023) benchmark on it: $u_t + uu_x = \nu u_{xx}$ on $[-1, 1] \times [0, 1]$ with $u(x, 0) = -\sin(\pi x)$, $u(\pm 1, t) = 0$ and $\nu = 0.01/\pi$. Their equation, their domain, their initial condition, their viscosity; no manufactured solution and no forcing. Salloum et al. (2026) evaluate on this same instance. Two registered implementation details are not met and they are named here rather than in a note: the run tag is not the one the registration names, and the initial and boundary conditions are enforced by a loss term over a fixed 90-point boundary set rather than by the registered ansatz $u = -\sin(\pi x) + t(1 - x^2) f_\theta(x, t)$, the network written $f_\theta$ here because $N$ is the pool size everywhere else, so part of the $R^2$ reported below is boundary error. Selection still touches only collocation points, which is what that contract item was for, and the treatment is the one every other PDE task here gets. Everything else is the protocol of Section 4 unchanged: $B = 40$, $N = 1{,}200$, $[64, 64]$ with $\tanh$, 4000 Adam steps, re-selection every 100, 64 shared seeds, one configuration signature with the other ladders. The reference solution is the Cole–Hopf quotient by Gauss–Hermite quadrature. Doubling the quadrature order moves it by

$1.6 \times 10^{-17}$ in relative $L^2$ on a $256 \times 100$ grid; it holds $u(\pm 1, t) = 0$ to $3 \times 10^{-16}$ over a hundred times, and has one interior zero at $x = 0$ with a layer that sharpens to width $0.013$ by $t = 0.5$. The $t = 0$ row is returned from the initial condition by a short circuit in the code and is therefore not a test of anything.

**This appendix exists because we registered two incompatible predictions about it and then ran it.** The first is a design note frozen before any of this was implemented. Its primary endpoint is *that the split appears*: every pointwise rule's slope negative and the set-level rule's not, at the sharpest qualifying viscosity. Its secondary endpoint is that a published pointwise rule crosses below uniform at high $V$ without already being worse at the smallest rung. Both are confirmed by the ladder below. The second is the prediction this paper's own Limitations carried, to justify excluding the task: that an optimisation-limited problem *suppresses* the damage, because a residual large everywhere does not localise, and therefore *no arm separates from uniform at any rung*. That one is refuted. The two contradict each other and both were on disk before the run. We report the confirmation and the refutation together, because reporting either alone would be a choice about which of our own registrations to honour.

**Nothing on this ladder is solving Burgers, and two floors say so.** Graded on the same $45 \times 45$ grid the runs are graded on, a predictor that ignores time entirely and returns the $t$-average $\bar{u}(x)$ scores $R^2 = 0.929$, and the frozen initial condition $u(x, t) = -\sin(\pi x)$, which evolves nothing and solves no equation, scores $0.664$. The whole ladder lies in $[0.447, 0.747]$: *no* cell reaches the time-independent floor and 25 of the 45 sit below the frozen initial condition. Neither floor is an artefact of that grid: on a $256 \times 256$ grid they are $0.927$ and $0.651$, because both predictors are smooth and the refinement barely moves them. Raissi et al. (2019) report $6.7 \times 10^{-4}$ relative $L^2$ on this instance and the re-drawn control's $0.71$ is about $54\%$. The grader is part of it: the layer is $2\nu/A \approx 0.006$ wide at $t = 0.5$ against a grid spacing of $0.045$, and the exact solution smeared in $x$ by a normalised $\mathrm{sech}^2$ kernel of half-width $0.10$, fifteen times too thick, still scores $0.966$, above every cell here.

A cell registered in advance at $5 \times 100$ with $B = 2000$, $N = 60{,}000$, $20{,}000$ Adam steps with cosine decay, the hard-constraint ansatz on, graded on $256 \times 256$, was gated on uniform reaching a relative $L^2$ below $0.15$, the line Rohrhofer et al. (2023) use to call a PINN run unsuccessful. It reached $\mathbf{0.284}$. **The gate fails**, so by that registration's own rule no $V$ axis is read there, and the four candidate ladder designs it enumerates are not run. Cost was not the obstacle: the largest of them fits in nine hours on seven cores.

A $2 \times 2$ over the boundary treatment and the grading grid says where the $0.284$ comes from, and two of its four cells share a training run so the grader can be isolated exactly. Read by two graders, one network scores $0.228$ on the $45 \times 45$ grid against $0.284$ on $256 \times 256$, so the coarse grid flatters relative $L^2$ by $1.24\times$: real, and far less than the smearing calculation above might suggest. Holding the grid fine and changing only the boundary treatment, the ansatz *helps* here, $0.284$ against $0.391$ for the penalty term, which is the opposite of its effect on `poisson1d` (Appendix O.1) and means the gate failure is not the ansatz's doing. Both fine-grid cells sit at or below the time-independent floor of $0.927$ on that grid. So scaling the network $25\times$ in parameters and the budget $50\times$, with five times the steps, **still does not beat a predictor that ignores time**.

**The one candidate explanation that registration named has been tried, and it is not the reason.** Every published run on this instance finishes with L-BFGS to reach $10^{-4}$, and the registration put it out of scope while naming it "the single most likely reason the gate fails". An amendment lifted that exclusion and changed nothing else, the gate least of all. Added after the Adam phase on the set the last gradient step used, L-BFGS converged on its own $10^{-11}$ tolerance at 2757 of a 5000 iteration cap and drove the residual on the selected points from $3.0 \times 10^{-5}$ to $1.4 \times 10^{-6}$, a factor of 21. The relative $L^2$ on the grading grid went from $0.284$ to $0.287$: **unmoved**. The amendment fixed that discriminator before the probe ran, and it reads the budget rather than the optimiser.

**And the budget can be read off the geometry, with no run at all.** The interior layer is $0.0129$ wide as the Cole–Hopf self-convergence check measures it and $2\nu/A = 0.0065$ analytically, and it runs the full extent in $t$, so it is $0.3$ to $0.7$ percent of the $(x, t)$ domain. A uniform draw of $B = 2{,}000$ therefore places about six to thirteen points inside it, while the spacing of that draw over a $2 \times 1$ domain is $\sqrt{2/2000} = 0.032$, which is $2.5$ times the layer's width. **The layer falls between the points.** Ten points cannot resolve a feature narrower than the distance between them, whatever the

optimiser does with them, which is also why a residual driven to $10^{-6}$ at those points says nothing about a layer none of them is in. At $B = 10{,}000$, the order published runs use, the same arithmetic gives thirty to sixty-five.

**What that costs and why we stopped.** Scaling this cell's measured per-arm costs, the authorised design at $B = 10{,}000$ needs about 178 hours on seven cores against the registration's 36-hour ceiling, the binding term being the set-level rule's pursuit at roughly $B^2$. Dropping that arm to afford the budget would remove the one the dividing line is about. So the operating point that would reach the published accuracy is not reached here, and the reason is recorded as a measurement rather than an intention.

So this appendix does not show the effect at the operating point this literature runs at. It shows a within-rule $V$ effect among configurations that all fail the benchmark, and the largest cell of Table 13, PINNACLE's $-0.26$ against the re-drawn control at $V/B = 20$, separation is a difference between degrees of not solving it. The within-rule comparison is paired on shared seeds and does not depend on the level, which is why we report it; the claim it cannot support is that a ladder at $B = 40$ and $R^2 \approx 0.7$ speaks for a ladder at $B$ in the thousands and $R^2 \approx 1$. It does not, and §7 carries that.

**What the ladder says.** Each arm against itself, smallest candidate set to largest, paired on the shared seeds:

| rule | paired mean $V/B : 1.25 \to 30$ | paired median | $p$ |
|---|---|---|---|
| BADGE | $-0.234$ | $-0.227$ | $3.4 \times 10^{-10}$ |
| PINNACLE | $-0.196$ | $-0.183$ | $8.2 \times 10^{-8}$ |
| RAD | $-0.145$ | $-0.141$ | $2.2 \times 10^{-4}$ |
| ours (kepiv) | $-0.126$ | $-0.119$ | $2.8 \times 10^{-4}$ |
| RAD+cov | $-0.100$ | $-0.087$ | $2.6 \times 10^{-4}$ |
| ours (addpiv) | $-0.092$ | $-0.066$ | 0.0027 |
| GRAD-MATCH | $+0.016$ | $+0.017$ | 0.30 |

Six of the seven degrade and the seventh is the set-level one.

**And five of the seven cross below the control, two of them under correction.** Failure counts are in Table 12 and the paired differences against the re-drawn control in Table 13. Against that control, at $9/64$, the first rung at which each arm is significantly worse by a two-sided Fisher exact test: BADGE and *our own* `kepiv` at $V/B = 5$ ($p = 0.001$ and $0.034$), PINNACLE at 20 ($p < 0.001$), RAD and *our own* `addpiv` at 30 ($p = 0.007$ and $0.034$). Holm over this panel's thirty-five Fisher tests leaves two, PINNACLE at $V/B = 20$ (adjusted $3.4 \times 10^{-4}$) and BADGE at 5 ($0.022$); RAD and both of our arms cross only uncorrected. `RAD+cov` never crosses although it degrades, and GRAD-MATCH never crosses and does not degrade. It sits at $7/64$ at each of the top three rungs and $8/64$ at $V/B = 1.25$, below the control at all four; at $V/B = 2.5$ it is $15/64$ against the control's $9/64$, above it on the count as well as on the paired statistic, so "below the control" is a four-of-five-rungs statement and Fisher there is $0.26$. On the paired statistic it dips once, $-0.067$ at $V/B = 2.5$ at $p = 0.046$, which is rank 24 of the 35 tests this ladder carries and nowhere near surviving Holm over them; it is the one cell on this task where the two readings disagree.

Both of our arms degrade in $V$ here and both end below the control. On `crenel` ours are the two arms that improve; on the field's own benchmark they are on the same side as the rules this paper criticises. The dividing line of Section 5 predicts exactly that, because our score is pointwise after the deflation and the objective is not a function of the selected set.

**The registered band, and why the releasing direction is not available here.** The registration behind this appendix, frozen before any of it was implemented, also asks for a viscosity band before any rule arm runs and for the opposite direction: a broad front at $\nu = 0.3/\pi$, where the second condition is released and the slope should flatten. The band is run, controls only, 64 seeds, and *its own gate disqualifies that direction*. The gate was registered as a failure count strictly inside $[3/64,\ 32/64]$ together with at least $0.05$ of headroom in mean $R^2$. The table reads it on the held control; the re-drawn one gives the same four verdicts, at $9/64$, $4/64$, $0/64$, $0/64$ and headroom $0.290$, $0.236$, $0.038$, $0.001$:

Table 12: `burgers1d` at $\nu = 0.01/\pi$: failures, seeds finishing below $R^2 = 0.5$ of 64. `uniform` is one held draw and `re-drawn` a fresh uniform $B$ at every exam; neither reads $V$. From `logEN_burgers1d_V{V}` by `scripts/vcurve.py --tasks burgers1d --prefix logEN`.

| $V$ | $V/B$ | uniform | re-drawn | RAD | RAD+cov | PINNACLE | BADGE | GRAD-MATCH | ours (addpiv) | ours (kepiv) |
|---|---|---|---|---|---|---|---|---|---|---|
| `burgers1d` | | | | | | | | | | |
| 50 | 1.25 | 7/64 | 9/64 | 8/64 | 5/64 | 6/64 | 6/64 | 8/64 | 10/64 | 12/64 |
| 100 | 2.5 | 7/64 | 9/64 | 7/64 | 17/64 | 14/64 | 11/64 | 15/64 | 10/64 | 12/64 |
| 200 | 5 | 7/64 | 9/64 | 13/64 | 14/64 | 19/64 | 27/64 | 7/64 | 12/64 | 20/64 |
| 800 | 20 | 7/64 | 9/64 | 19/64 | 11/64 | 33/64 | 34/64 | 7/64 | 18/64 | 24/64 |
| 1200 | 30 | 7/64 | 9/64 | 23/64 | 11/64 | 32/64 | 33/64 | 7/64 | 20/64 | 20/64 |

| $\nu$ | uniform fails | headroom | verdict |
|---|---|---|---|
| $0.01/\pi$ | $7/64$ | $0.253$ | qualifies, and is the ladder above |
| $0.03/\pi$ | $4/64$ | $0.210$ | qualifies, and is the ladder below |
| $0.1/\pi$ | $0/64$ | $0.090$ | fails the band |
| $0.3/\pi$ | $0/64$ | $0.015$ | fails both |

**And the rung that qualified has now been run.** It had stood in this table as "qualifies, not run" since the band was read. The registration fixes the ladder, the arms and both endpoints, so this is its execution and not a new design. The band holds at the ladder's own seeds, $5/64$ held and $4/64$ re-drawn at every rung with $0.766$ of held mean $R^2$. Those are the ladder's numbers, at $\nu$ written to fifteen digits. The band row that qualified this rung was read at the seven-digit decimal and gives $4/64$ and $0.790$ instead. Two encodings of the same viscosity, one seed and $0.024$ of headroom apart, and no verdict between them: that is the instance sensitivity this appendix warns about below, measured here rather than asserted. The primary endpoint confirms: within-rule from $V/B = 2.5$ to 30, RAD $-0.044$ ($p = 0.043$), PINNACLE $-0.188$ ($2.9 \times 10^{-7}$), BADGE $-0.212$ ($5.2 \times 10^{-8}$), and GRAD-MATCH $+0.061$ ($2.1 \times 10^{-4}$). Every pointwise slope negative, the set-level one positive. Our own two arms degrade here too, $-0.105$ and $-0.101$, which is the second task on which they do.

The *secondary* endpoint, registered as tested and not promised, also fires: against the re-drawn control at $4/64$, PINNACLE reaches $21/64$ and $22/64$ and BADGE $21/64$ and $21/64$ at the top two rungs, all four surviving Holm over this panel's thirty-five tests, while GRAD-MATCH runs 2, 3, 3, 0 and 2 failures of 64 across the five rungs and never crosses. **Five of the seven arms cross uncorrected and two of the five are ours**: RAD at $13/64$ at $V/B = 20$ ($p = 0.035$), and our own `addpiv` at $15/64$ and `kepiv` at $17/64$ at the last rung ($p = 0.011$ and $0.003$), none of the three surviving Holm over the thirty-five. That is the same pattern as the $\nu = 0.01/\pi$ ladder, which this paper states in full, so the replication is closer than the four-arm sentence above makes it sound, and the half of it we had not printed runs against our own arms. **This is a replication and not a new task**: the split and the crossing are already on `burgers1d` at $\nu = 0.01/\pi$, so no count in this paper moves. What moves is that neither rests any longer on the single viscosity the field's own instance fixes.

At the two broad viscosities uniform never fails, and at $0.3/\pi$ there is one and a half percent of headroom left in $R^2$, so nothing there is winnable or losable. That is the two-sided void rule of Section 2, and it says the releasing direction cannot be measured on this family at this capacity. It was applied before any arm ran *there*, which is the only place it matters and is less than the registration asked: the registered order was the whole band before any rule arm anywhere, and what gated the sharp rung was instead a 16-seed two-sided void probe at that viscosity alone, with the full band run after the ladder. It is measured on three others (Table 2). So this appendix shows the effect where the conditions hold and does not show it disappearing where they do not.

**One thing this instance taught us about the instance.** The band's sharpest rung was first run with $\nu$ written as the decimal $0.0031831$ rather than as $0.01/\pi$, a relative difference of $3.6 \times 10^{-7}$. The harness is otherwise exactly reproducible: re-running that rung at $\nu = 0.01/\pi$ reproduces the ladder's two controls on all 64 seeds to the last digit. With the decimal, individual seeds move by up to $0.53$ in $R^2$ and the uniform failure count moves by 2 of 64. Every comparison in this appendix is paired within one instance on shared seeds and none of them is affected. What it does say is that a comparison *between papers* on this benchmark, at the three to ten seeds this literature reports, is measuring less than it appears to.

Table 13: `burgers1d` at $\nu = 0.01/\pi$: paired mean of (rule − re-drawn control) in test $R^2$ over the same 64 seeds. Bold is a two-sided Wilcoxon $p < 0.05$. The held control's row is what re-selection is worth on this task, which is nothing measurable. From the same runs, by `scripts/vcurve_paired.py` against `random_resample_exam`.

| | | $V/B = 1.25$ | $V/B = 2.5$ | $V/B = 5$ | $V/B = 20$ | $V/B = 30$ |
|---|---|---|---|---|---|---|
| `burgers1d` | RAD | −0.008 | −0.038 | **−0.081** | **−0.141** | **−0.154** |
| | RAD+cov | +0.018 | **−0.084** | **−0.064** | **−0.084** | **−0.082** |
| | PINNACLE | −0.025 | **−0.098** | **−0.156** | **−0.263** | **−0.220** |
| | BADGE | −0.008 | **−0.108** | **−0.194** | **−0.246** | **−0.241** |
| | GRAD-MATCH | −0.023 | **−0.067** | −0.013 | +0.004 | −0.007 |
| | ours (addpiv) | −0.045 | **−0.065** | **−0.067** | **−0.149** | **−0.136** |
| | ours (kepiv) | −0.051 | **−0.079** | **−0.118** | **−0.181** | **−0.177** |
| | uniform (held) | +0.037 | +0.037 | +0.037 | +0.037 | +0.037 |

## N AT EQUAL COMPUTE, AGAINST UNIFORM SELECTION WITH MORE POINTS

`logDB_{task}_s100` gives every arm at $B = 40$ with its FLOP count, `logDG_{task}_B{20,...,320}` gives uniform selection across the budget at the same $N$, the same 4000 steps, the same cadence and the same network, and uniform at $B = 40$ reads 4.06 GFLOP in both, so the accounting is one accounting. The six arms are every selector Table 1 carries except our `addpiv`. Putting it back changes no verdict here: it prices identically to `kepiv` at 9.30 GFLOP and needs the same 1.74× uniform budget, so the count reads three of seven and the three are the same three.

Table 14: At equal compute: each arm at $B = 40$ against uniform selection at a larger budget. Mean test $R^2$, $N = 1{,}200$, 4000 steps, $[64, 64]$ with tanh, the uniform rows from `logDG_{task}_B{20,40,80,160,320}` at 32 seeds and the arm rows from `logDB_{task}_s100` at 100. **The two seed counts differ and the comparison is unpaired**: on `poisson1d` uniform at $B = 40$ reads −1.94 over logDG's 32 and −3.02 over logDB's 100. Restricting the arm rows to the ladder's own 32 seeds moves one verdict of six, PINNACLE's matching budget from $B = 80$ to $B = 160$ and its ratio from 0.24× to 0.48×; the other five are unchanged and the three dominated arms stay dominated. The control here is a *held* draw; Table 15 is the same computation against the re-drawn one. Generated by `scripts/isoflop.py`.

| arm | $B$ | GFLOP | `crenel` | `poisson1d` |
|---|---|---|---|---|
| uniform | 20 | 2.03 | 0.472 | −10.368 |
| uniform | 40 | 4.06 | 0.794 | −1.937 |
| uniform | 80 | 8.11 | 0.925 | 0.692 |
| uniform | 160 | 16.22 | 0.954 | 0.996 |
| uniform | 320 | 32.44 | 0.978 | 0.999 |
| RAD | 40 | 5.27 | 0.717 | −0.667 |
| RAD+cov | 40 | 5.27 | 0.802 | 0.888 |
| PINNACLE | 40 | 33.65 | 0.445 | 0.022 |
| BADGE | 40 | 32.93 | 0.577 | −0.030 |
| GRAD-MATCH | 40 | 32.93 | 0.940 | 0.976 |
| ours (kepiv) | 40 | 9.30 | 0.946 | −0.304 |

**Three of six arms are dominated by a *held* uniform control, and they are the three most expensive.** Against the re-drawn one it is four, which is the number Section 3 quotes and the re-drawn frontier below derives. Taking the cheapest uniform budget that matches or beats an arm on *both* tasks: PINNACLE at 0.24× its compute, BADGE at 0.25×, GRAD-MATCH at 0.49×. Uniform selection with four times the points is better than GRAD-MATCH on both tasks at half of GRAD-MATCH's compute, and GRAD-MATCH is the rule this paper recommends.

**Three survive, and they are the three cheapest.** Uniform needs 1.54× RAD's compute to match it, 1.74× ours, and 3.08× RAD+cov's. The frontier at this scale is held by the cheap rules, which is the same thing Section 3 says when it reports that a one-line coverage floor on RAD beats our own arms

at a quarter of the cost. The split is not a rounding artefact: the three surviving ratios start at $1.54$ and the three dominated ones stop at $0.49$, with nothing in between on the mean under the held control, so a factor of three separates the two groups there. On the median, and against the re-drawn ladder, GRAD-MATCH does sit at the boundary; the paragraph on the change of statistic below says so.

**Every ratio here is a *sampled* budget, and the grid is coarse.** The uniform ladder runs $B \in \{20, 40, 80, 160, 320\}$, so a ratio is the cheapest budget *we ran* that matches or beats an arm on both tasks, not the cheapest that exists. Where the two surrounding rungs straddle an arm the true crossing is somewhere between them and we did not look: against the re-drawn control `RAD+cov` is not matched at $B = 40$ ($0.77\times$; $0.761$ against its $0.888$ on `poisson1d`) and is at $B = 80$ ($1.54\times$), so anything in $(0.77, 1.54]$ is consistent with these runs. The main text's claim is therefore stated as no *sampled* uniform budget matching it below $1.54\times$, which is what the ladder supports. The direction of the error is known: a finer grid can only lower a surviving ratio and raise none, so the three dominated arms stay dominated and the three survivors could survive by less.

**The median is a flag on the same script, not a separate analysis.** `scripts/isoflop.py --stat median` recomputes the whole frontier from the same `per_seed.csv` files on the same seeds, and the four verdicts quoted below are its output rather than a figure computed once by hand.

**The uniform in Table 14 is a held draw too, and re-drawing it moves four of the six verdicts.** That frontier compares each arm against `random_subset` at a larger budget: one draw at step $0$, held for $4000$ steps, while every arm re-selects $41$ times. Appendix O measures what that costs on `crenel` at $B = 40$; here it is measured at every budget on the same $32$ seeds, with the held column reproducing `logDG`'s exactly in all ten cells, so the two run tags are one experiment. Re-selection is worth a factor of two to four *in budget* at the cheap end and nothing above it: on `poisson1d` the held control fails $25/32$ at $B = 20$ and $15/32$ at $40$ where a re-drawn one fails $9/32$ and $1/32$, and by $B = 80$ they are $2/32$ and $0/32$.

Table 15 is the same computation against the re-drawn ladder. RAD changes side: uniform at $B = 20$, half the points and $0.38\times$ the compute, matches it on both tasks. PINNACLE and BADGE fall from $0.24\times$ and $0.25\times$ to $0.06\times$. GRAD-MATCH and our arm do not move at all, because their matching budget is $B = 160$, where the two controls agree. Four of six arms are beaten by uniform at no more compute rather than three. The two that survive are *not* the two cheapest: RAD ties RAD+cov as the joint-cheapest arm, at $5.27$ GFLOP each, and is the one that changes side.

**One verdict here does not survive a change of statistic, and it is not one of those.** Table 14 reports the mean $R^2$, and on `poisson1d` at $B = 20$ that mean is $-10.4$, where Table 15's re-drawn control reads $+0.359$ in the same cell, so a reader is entitled to ask for the median. On the median the flip is the same shape: RAD $1.54\times \to 0.77\times$, RAD+cov $3.08\times \to 1.54\times$, PINNACLE and BADGE $0.24\times$ and $0.25\times \to 0.12\times$. What changes is GRAD-MATCH, which reads $0.49\times$ on the mean and $0.99\times$ on the median under *either* control. It sits at the boundary rather than comfortably dominated, and the sentence above that has uniform with four times the points beating GRAD-MATCH at half its compute is a mean-only statement.

$B = 320$ is a quarter of a $1{,}200$-point pool, and Section 2 already reports that selection stops changing whether a run fails by $B \approx 80$. So the frontier is partly the statement that at our pool size one can simply buy the pool, which is exactly the regime the paper says its claims do not concern.

## O THE UNIFORM CONTROL IS A SINGLE DRAW, AND WHERE THAT MATTERS

**The difference, stated plainly.** Every rule in this paper's protocol re-selects at every exam, $41$ times in a run. The uniform control does not: `random_subset` draws $B$ points once at step $0$ and the network trains on those points for all $4000$ steps. So the arms differ in more than the rule, and Section 2 now lists it as the first of its three exceptions.

**The two controls separate on one of Table 1's two tasks, not on both.** On `poisson1d` the held draw fails $52/100$ and the re-drawn one $1/64$, a Fisher $p = 1.6 \times 10^{-13}$. On `crenel` it is $3/100$ against $0/64$, Fisher $p = 0.28$: under a $3\%$ rate the chance of seeing none in $64$ draws is $0.14$, so that row is consistent with the two controls being the same thing and establishes nothing by itself. The case for re-drawing rests on `poisson1d` and on the ladder, where the choice of control moves

Table 15: The frontier of this appendix recomputed against a uniform control *re-drawn* at every exam, which is the schedule every arm runs on, instead of one held from step 0. Mean test $R^2$; the arm rows are unchanged from Table 14. Generated by `scripts/isoflop.py --ctrl-tag logEO --ctrl-method random_resample_exam`.

| arm | $B$ | GFLOP | `crenel` | `poisson1d` |
|---|---|---|---|---|
| uniform | 20 | 2.03 | 0.756 | 0.359 |
| uniform | 40 | 4.06 | 0.836 | 0.761 |
| uniform | 80 | 8.11 | 0.920 | 0.983 |
| uniform | 160 | 16.22 | 0.948 | 0.999 |
| uniform | 320 | 32.44 | 0.972 | 0.999 |
| RAD | 40 | 5.27 | 0.717 | −0.667 |
| RAD+cov | 40 | 5.27 | 0.802 | 0.888 |
| PINNACLE | 40 | 33.65 | 0.445 | 0.022 |
| BADGE | 40 | 32.93 | 0.577 | −0.030 |
| GRAD-MATCH | 40 | 32.93 | 0.940 | 0.976 |
| ours (kepiv) | 40 | 9.30 | 0.946 | −0.304 |

which rules cross, and not on `crenel`'s row, which we report because it is part of the table rather than because it argues for us.

**And three of Table 1's five tasks have no re-drawn row at all.** The reason is scope: the control exists at 64 seeds on the two tasks Section 4's ladder runs, and the ladder chose the tasks with room for a rate test. That is not a scientific reason, and it is mildly circular, because the three without a row are exactly the three where the held control already fails almost nothing, $9/100$, $1/100$ and $3/100$. Those are the cells where re-drawing has the least to show and the least to be needed, and we cannot report what they would say.

**And the fix is not ours; it is in two of the papers we benchmark.** Wu et al. (2023) run "Random sampling with resampling (Random-R) ... the same as the Random method, except that the residual points are resampled for every $N$ iteration", report that "the Random-R method always leads to lower $L^2$ relative errors regardless of the period of resampling" and that "the error can be lower by one order of magnitude", conclude that "among all the non-adaptive methods, Random-R performs the best", and list testing it among their own first-time contributions. They also note it is RAD at $k = 0$ or $c \to \infty$, so the held control is not a member of RAD's family and the re-drawn one is its degenerate member. Daw et al. (2023) define the same pair, "(1) fixed: which uses a fixed set of uniformly sampled collocation points, (2) dynamic: a simple modification where the collocation points are dynamically sampled from a uniform distribution every iteration", and report the dynamic one converging where the fixed one fails. **So this appendix is not reporting an unused control.** It is reporting that a published control exists, that we used the other one, and that the choice decides which rules cross. The measurement below is ours; the control is theirs.

**It is measurable, because the harness already carries the other control.** `random_resample_exam` re-draws $B$ uniform points at every exam: the same schedule as every rule, no score, no selection cost. The runs that carry both predate this study and are not all at the protocol above, so the table names each one's, and the counts are over *distinct* seed-runs: a held control cannot depend on the re-selection cadence, so among the `vburgers1d` directories two share one held column and a third is contained in a fourth verbatim.

| task family | one draw, held | re-drawn each exam | runs |
|---|---|---|---|
| `crenel` | $2/64$ | $0/64$ | $B$=40, 4000 steps, cadence 100; five rungs |
| `poisson1d` | $10/20$ | $2/20$ | $B$=40, 4000 steps, cadence 100; four pools |
| `vburgers1d` | $0/8$ | $0/11$ | $B$=40, 4000 steps, cadences 100 and 500 |
| `poisson2d` (ring) | $14/15$ | $14/15$ | $B$=120 and 400, 8000 steps, cadence 200 |
| `transport1d` | $0/3$ | $0/3$ | $B$=40, 4000 steps, cadence 100 |

**The effect is measured on `poisson1d`, and the other three rows are weaker evidence than they look.** A frozen sparse draw resolves whatever it happens to land on and nothing else, so it is worst where the solution has structure too narrow for $B$ uniform points to catch reliably, which is the

static sharp interior layer: the held draw fails half the time where a re-drawn one fails twice in twenty. The rest of the table cannot carry "and nowhere else", by this paper's own rule. On `vburgers1d` and `transport1d` the control fails $0/n$, which Section 2 calls **void**: nothing is winnable there and their 0 against 0 is an absent measurement, not a measured absence. What those two do say is on the graded statistic, and it is not quite nothing: over the distinct seed-runs, re-drawing wins 8 of 11 on `vburgers1d` at a median of $+0.008$, in a spread from $-0.21$ to $+0.13$, and 3 of 3 on `transport1d` at a median of $+0.005$ with nothing above $+0.016$. A consistent sign at a hundredth of $R^2$, against the $+0.43$ median `poisson1d` shows. The ring is worse than void: both controls fail 14 of 15, and its paired median is $-2.352$ with an interquartile range of $[-7.79, -0.21]$, so on that task the re-drawing control is the worse of the two and the cell settles nothing in either direction.

`poisson1d` is one of the two tasks in Section 4's sweep, so the claims there that are stated *against the control* are measured against a control that a re-drawing uniform baseline beats. Two of the three claims this used to qualify are gone rather than qualified: Appendix L retires "every published rule helps at every $V$" and "none crosses below the control" outright, since RAR-G is published and crosses there. What survives is the $+2.2$ to $+4.6$ band, which is inflated by an unknown amount and should be read as against a held draw. The within-rule claims on the same panel are untouched, because no control enters them: best rung against worst, the factor of 3.3 to 5.5, and our own arm ending negative.

*The experiment this appendix registers has been run at all five rungs on the same* 64 *seeds*: `random_resample_exam` beside the published rules, same pool, same schedule, same protocol. It is verifiable that the two directories are one experiment, because the held control reproduces across them to the last digit on every seed.

Re-selection alone is worth $+0.049 \pm 0.046$ in mean $R^2$ on `crenel` ($p = 0.025$, paired Wilcoxon over 64 seeds), and the re-drawn control fails $0/64$ where the held one fails $2/64$.

**Second, what it does to the sweep.** Table 16 is the full ladder read against the re-drawn control, which is the *right* control for this claim and not only the harder one. The crossings do not move: PINNACLE and BADGE still cross at $V/B = 5$ and RAD at 20, at Fisher $1.9 \times 10^{-6}$, 0.0013 and $6.1 \times 10^{-4}$. RAD's is the rung that depends on the test: at $V/B = 5$ it is $-0.043$ at Wilcoxon $p = 0.068$ and Fisher 0.50, both short of the line, where a paired $t$-test gives 0.026 and would move its crossing from 20 to 5. The rung reported is the one the sweep's registered test gives, and Appendix J is where that choice is varied. The reversal is larger at the far end, because the bar is higher everywhere: $-0.300 \to -0.350$, $-0.169 \to -0.218$, $-0.050 \to -0.099$. GRAD-MATCH and both of our arms keep their sign and their significance at every rung. What the honest control takes is the *start*: PINNACLE's $+0.077$ becomes $+0.028$ and is no longer significant, so on this task its curve reads "level with re-drawing uniformly, then worse", not "better, then worse". BADGE's $+0.083$ becomes $+0.034$ and survives at $p < 0.05$. One direction is not measurable on this panel at all and the table must be read knowing it: at $0/64$ the control cannot be beaten on the failure count, so that column can show harm and never help, and every positive claim here rests on the paired statistic, which is where it already rested.

**What a reader outside this paper should take from the appendix.** The finding here is not about our control; it is about the baseline this literature uses. A uniform baseline drawn once and held is not the same experiment as an adaptive rule that re-selects, and the difference is large exactly where adaptive sampling is interesting. It is not always present, either: on `burgers1d` the difference is $+0.037$ at $p = 0.20$, which is the held control nominally ahead and indistinguishable from no difference at all.

**Third, the arm that was missing from every ladder.** `rad_cover_exam`, RAD with half the budget reserved for uniform coverage, is the cheapest rule that survives the iso-FLOP frontier of Appendix N and it had never been run on a $V$ axis. It never crosses ($-0.048$, $p = 0.10$ at the last rung) and it is never significantly *above* the re-drawn control either, at any rung. Both halves are the same fact and it is the one this paper predicts: a coverage floor buys immunity to $V$ by making the rule more nearly uniform, and what it buys is paid for in the ceiling.

Table 16: `crenel`, paired mean of (rule − control) in test $R^2$ over the same 64 seeds, against the *re-drawn* uniform control rather than the held one. Bold is a two-sided Wilcoxon $p < 0.05$. The held control's own row is what re-selection alone is worth, and it is one measurement printed five times: neither control reads $V$, so those five cells are the same 64 paired differences. A uniform draw from a uniform window of $V$ is a uniform draw, and the five rungs agree to the last digit in every outcome column. Generated by `scripts/vcurve_paired.py --prefix logDF logEN logET --tasks crenel --ctrl-method random_resample_exam`.

| | | $V/B = 1.25$ | $V/B = 2.5$ | $V/B = 5$ | $V/B = 20$ | $V/B = 30$ |
|---|---|---|---|---|---|---|
| `crenel` | RAD | $+0.021$ | $-0.014$ | $-0.043$ | $\mathbf{-0.126}$ | $\mathbf{-0.099}$ |
| | RAD+cov | $+0.012$ | $+0.014$ | $-0.022$ | $-0.026$ | $-0.048$ |
| | PINNACLE | $+0.028$ | $+0.008$ | $\mathbf{-0.208}$ | $\mathbf{-0.369}$ | $\mathbf{-0.350}$ |
| | BADGE | $\mathbf{+0.034}$ | $+0.011$ | $\mathbf{-0.142}$ | $\mathbf{-0.294}$ | $\mathbf{-0.218}$ |
| | GRAD-MATCH | $\mathbf{+0.041}$ | $\mathbf{+0.102}$ | $\mathbf{+0.122}$ | $\mathbf{+0.128}$ | $\mathbf{+0.137}$ |
| | ours (addpiv) | $\mathbf{+0.096}$ | $\mathbf{+0.107}$ | $\mathbf{+0.113}$ | $\mathbf{+0.119}$ | $\mathbf{+0.118}$ |
| | ours (kepiv) | $\mathbf{+0.093}$ | $\mathbf{+0.100}$ | $\mathbf{+0.122}$ | $\mathbf{+0.136}$ | $\mathbf{+0.140}$ |
| | uniform (held) | $\mathbf{-0.049}$ | $\mathbf{-0.049}$ | $\mathbf{-0.049}$ | $\mathbf{-0.049}$ | $\mathbf{-0.049}$ |

### O.1 The boundary condition enforced exactly: the control run, the ladder gated out

Two blind readers raised the same objection independently and gave the same mechanism for it. `poisson1d` is $u'' = f$, linear and second order, so its solution is pinned only up to an affine function without boundary data; its Dirichlet set is two points at $\lambda_{bc} = 1$ against a forcing with $\mathrm{mean}(f^2) = 4462$, so that term is outweighed by roughly 4000 to 1; and $\mathrm{Var}(u^*) = 0.1015$, so a constant offset of 0.225 alone fails our threshold. Their conclusion is that what this paper reads as selection damage on that task may be largely free drift in the affine mode.

**What exists is the fix, not the answer.** Replacing the network by $u_\theta(x) = (1-x)u^*(0) + x\,u^*(1) + x(1-x)N_\theta(x)$ makes the boundary residual identically zero, so the affine mode cannot exist, and the boundary term is then dropped because there is nothing left for it to penalise. Selection still indexes only into the collocation pool. The transform is implemented and verified pointwise: it returns $u^*(0)$ and $u^*(1)$ to machine precision at the two boundary points.

**Stage 0, and the objection's mechanism runs the wrong way.** Before any rule arm sees the ansatz we read the uniform controls at 64 seeds. The held control fails $\mathbf{53/64}$, mean $-21.6$, median $-4.12$; the re-drawn control fails $2/64$. Under the penalty term the same held control fails $35/64$ at a median of $+0.28$. So enforcing the condition exactly makes the held control *worse*, and the objection predicts the opposite: if those failures were largely free drift in the affine mode, removing the mode by construction should have removed most of them. Whatever they are, they are not that.

**And the ladder is not read, by a threshold fixed before the run.** Every cell in this paper must have a uniform control failing strictly inside $[3/64, 32/64]$, so that something is winnable and something losable. At 0.828, with a Wilson lower bound of 0.718, this one is outside; under the ansatz `poisson1d` at $B = 40$ is a harder problem than the one the rest of the paper measures, not the same problem with a confound removed. Reading a $V$ ladder there would be choosing the band after seeing the answer.

**One thing it does settle.** The gap between the two uniform controls *widens* under exact enforcement, $53/64$ against $2/64$, so the re-drawing result of Appendix O does not depend on how the boundary is treated.

So this appendix reports an objection whose stated mechanism fails and which is otherwise open. Every `poisson1d` claim in this paper is stated against the penalty boundary term, and Section 7 lists it among the confounds not removed. A ladder at a budget where the hard-constraint control lands inside the band would settle it; that needs a larger $B$ and a new registration.

**Why the first attempt here was void, recorded rather than quietly replaced.** The flag enabling the ansatz was set on the main process's configuration object, and the seed sweep spawns its workers, so each worker re-imported the module, re-read the configuration from file, and trained with the penalty term as before. The output was bit-identical to the penalty ladder on all 64 seeds and every

arm, which is how it was caught: a run that changes nothing changes nothing exactly. The metadata did not catch it because it was written from the main process's view rather than the workers'; it now records what the workers ran and refuses to write when they disagree.

## P BASELINE IMPLEMENTATIONS, AND HOW FAR EACH IS FROM THE RULE THE IDENTITY DESCRIBES

**The one partial exception to "none reports the size".** Nabian et al. (2021) do report and sweep a size: the number of *seeds* at which the loss is approximated before importance sampling, $10{,}000$ in their main example, $S = 50$ as a small case, and they draw the conclusion opposite to this paper's title: "the number of seeds, if selected reasonably (e.g. $S \geq 500$ in this case), does not significantly affect the convergence behavior and therefore there is no need to consider that number as a new hyperparameter."

It is not the knob of this paper, and the difference is structural rather than a matter of degree. Their seeds set the resolution at which a sampling *density* is approximated, and every collocation point is then drawn from that density. Nothing is ranked, so there is no $\arg\max$, no top-$B$, and no quantile for a count to move.

Every baseline is reimplemented as a selection rule inside the same harness, so two arms differ only in the rule and share the network, the optimiser, the data, the seed and, among the rules, the re-selection schedule. RAD follows $p(x) \propto \epsilon(x)^k / \overline{\epsilon^k} + c$ with the authors' robust default $k = c = 1$ (Wu et al., 2023). R3 keeps points above the mean residual and re-draws the rest (Daw et al., 2023). PINNACLE scores $\hat{\alpha}(z) = r(z)^2 k_z^\top K_{\mathrm{ref}}^{-1} k_z$ and selects by $k$-means++ (Lau et al., 2024). Lau et al. give two variants and every PINNACLE cell here is the $k$-means++ one, which the harness names `pinnacle_kmeans_exam`. The sampling variant exists in the harness, is reported nowhere in this paper, and is what would separate the score from the selection step built on it. **And there are two departures from PINNACLE as published, not one**: the pool size is the parameter under study, and we fix the boundary set rather than letting the rule allocate budget jointly across point types, which is the part of its contribution this study does not exercise. GRAD-MATCH runs orthogonal matching pursuit on atoms $a_j = r_j R_j$ against the target $\sum_j r_j R_j$ (Killamsetty et al., 2021). BADGE runs $k$-means++ on gradient embeddings $g_x = r_x R_x$ (Ash et al., 2020). $R$ is a Nyström factor (Williams & Seeger, 2001; Mahoney, 2011) of the tangent features throughout, with $m = 64$ landmarks everywhere, for BADGE and GRAD-MATCH through $R$, for PINNACLE through a 64-point reference set, and for our two through the same factor on their window; RAD and RAD+cov build none. **The landmark *fraction* is not shared and the approximation is.** In Table 1 those 64 landmarks carry $1{,}200$ rows for the others and 200 for ours, $5.3\%$ against $32\%$. That sounds like an advantage and is not one, because the factor's rank is set by the $m \times m$ landmark Gram and not by the pool, and it comes out the same at either size: the factor keeps 33–50 directions of a possible 64 at both, at a relative Frobenius error of at most $7.4 \times 10^{-6}$ on `poisson1d` and $3.8 \times 10^{-3}$ on `crenel` (`scripts/nystrom_share.py`, both tasks, at initialisation and after 300 steps, three seeds each). **The smaller pool is not the better approximation**: the window is the worse of the two in eight of those twelve cells, and the largest error of the twelve is the window's. Those kept counts are what the harness's `float32` features retain, and at initialisation most of it is round-off. Forming the same $64 \times 64$ landmark Gram from the same rows in single and in double precision and applying `nystrom_factor`'s own $10^{-10}\lambda_{\max}$ cut to each, the single-precision one keeps 33–36 directions at initialisation where the double-precision one keeps 6 on `poisson1d` and 11 on `crenel`; after 300 steps it is 41–49 against 24–40. So four fifths of the single-precision rank at initialisation is arithmetic noise, and about a quarter of it after training. Two numbers an earlier version of this paragraph took from a docstring rather than from a run are corrected with it: the double-precision count of 8 is the rank of the *feature matrix*, a different object that `scripts/nystrom_share.py` also reports; and the smallest retained eigenvalue of the single-precision Gram is not negative at $-6.6 \times 10^{-5}$ but positive, between $1.0 \times 10^{-7}$ and $2.9 \times 10^{-6}$ over the twelve cells. All four counts and both minima go to `summary.csv` now. Either reading gives the same conclusion, and the double-precision one gives it more strongly, since a rank of 6 to 11 needs far fewer than 64 landmarks at any pool size. So the asymmetry Section 2 discloses is about which points are scored.

**The baselines carry no retention and our arms do.** The `win:` wrapper rebuilds a baseline's whole selection at every re-selection; our two arms hold a first-in first-out tail and carry over half the budget each round. So Table 1 and the ladder compare our method against theirs, not one ingredient against itself, and R3 is the one published rule that also retains, which is why it is excluded above.

**What the retention exception does to the truncation ratio.** Our arms replace $b = B/2 = 20$ points per round where a rebuilt baseline replaces all $B = 40$, so a common $V$ is not a common pressure: ours keeps the best 20 of $V$ and theirs the best 40 of $V$. Every rung is twice the truncation ratio for our arms that it is for a rebuilt baseline. The ladder tables and Figure 2 are labelled in $V/B$ with $B = 40$ throughout, so our two rows are labelled at half the ratio they select at: our $V/B = 5$ column truncates as a rebuilt rule's 10 would. At the first two columns doubling lands exactly on the next rung, the spacings being 2, 2, 4 and 1.5, and at the other three it lands between rungs or off the top. One of the two adds a difference of another kind. The `add` arm scores point $i$ in a window by $(1 - w)\, e_i / \max_j e_j + w\, r_i / \max_j r_j$ with $w = 1/2$ (the harness default, in `config.yaml` at each run's recorded commit; runs made after this was noticed record it in their own metadata), both maxima over the window, so the weight geometry carries against the residual depends on which other points the draw contained and the score is not a fixed function of the point. Proposition 3 assumes one, so this is a second reason that arm sits outside the identity, beyond the softening this appendix is about.

**The names in the shipped runs.** The supplementary carries the per-seed files, whose `method` column uses the harness's names and not the paper's. The map, and it is the whole of it. The `win:` prefix below is the ladder's form: in Table 1's own files the same five rules appear *bare*, which is the asymmetry Section 2 discloses, visible in the file names themselves:

| in the paper | in `method` | |
|---|---|---|
| uniform control | `random_subset` | one draw at step 0, held |
| re-drawing control | `random_resample_exam` | $B$ fresh uniform points every exam |
| RAD | `win:rad_exam` | |
| PINNACLE | `win:pinnacle_kmeans_exam` | the $k$-means++ variant |
| BADGE | `win:ntk_badge_exam` | fixed budget; published form grows its set |
| GRAD-MATCH | `win:ntk_gradmatch_exam` | |
| ours (addpiv) | `ntk_addpivoted_snake_exam` | no `win:`; windows internally |
| ours (kepiv) | `ntk_kepiv_snake_exam` | no `win:`; windows internally |
| RAD+cov | `win:rad_cover_exam` | our coverage floor on RAD |
| RAR-G | `win:rar_g_exam` | fixed budget; `rar_grow_exam` grows |
| dynamicAL | `win:ntk_dynamical_exam` | fixed budget; published form grows its set |
| growing uniform | `random_grow_exam` | control for the growing arms |

`win:`$R$ is the wrapper of Section 2; the same name without it is the rule on the whole pool, which is what Table 1 reports for the published arms. `_exam` marks a rule that re-selects at every exam rather than once. Files under `runs/` also carry arms this paper does not report, from sweeps that predate it.

**No *softened* arm here keeps the best $B$ of $V$, and two of them are one score.** The identity this paper is built on is stated for a top-$B$ rule, and Propositions 3 and 4 assume one. Of the eight arms on the ladder, RAR-G and dynamicAL are top-$B$ rules and this appendix is about them; the other six are not, and they fail to be one in two different ways. RAD samples without replacement from a density (Appendix I gives the weaker form in which the quantile still reaches it, and says in as many words that it is not a quantile for RAD). GRAD-MATCH is orthogonal matching pursuit and is excused separately, having no quantile for $V$ to move at this budget. **At twice it the excuse already fails, and at four times it fails further**: on `poisson1d` GRAD-MATCH's failures rise with $V$ from $1/32$ to $8/32$ at $B = 80$ and from $0/32$ to $7/32$ at $B = 160$, where its paired mean against the control is $-2.44$ ($p = 2.0 \times 10^{-5}$), so the arm this paper recommends is harmed by the candidate set once the budget is large enough (Appendix X). PINNACLE and BADGE are $k$-means++: *one* $\arg\max$ over the window, then $B - 1$ picks drawn proportional to squared distance from what is already chosen. Only the first of forty picks is a best-of-$V$.

**And GRAD-MATCH is not spared by having no $\arg\max$; it takes $B$ of them.** Each of its $B$ steps picks $\arg\max_j |a_j^\top \rho| / \|a_j\|$ over the window, so the main text's "no quantile for $V$ to move" was

loose. What it has that our arms do not is that $\rho$, the part of the pool gradient not yet matched, is *re-fit by least squares after every pick*: nothing a point is scored by at pick 1 is what it is scored by at pick 40. Our arms deflate the geometry the same way and carry the residual factor $r_j$ unchanged through all twenty picks of a round, so a per-point score does survive there. It is a description and not a proof: we have measured that GRAD-MATCH does not degrade at $B = 40$ and have not shown that re-fitting is why.

**Nor is it covered by the $(1 - 1/e)$ guarantee, and its own guarantee is vacuous for what we run.** Killamsetty et al. (2021) do not claim submodularity for the gradient-matching objective. Their Theorem 2 gives $\gamma$-weak submodularity with $\gamma \geq \lambda/(\lambda + k\nabla^2_{\max})$, and the corollary gives OMP a $1 - \exp(-\lambda/(\lambda + k\nabla^2_{\max}))$ approximation, where $\lambda$ is the regularisation on the weights and $k$ the budget. At $\lambda = 0$ that bound is 0. Our `gradmatch_select` re-fits the weights by unregularised least squares, so $\lambda = 0$ and the guarantee we would be invoking says nothing about the arm we ran.

So for the two rules that carry the crossing, $V$ reaches the selection the way it reaches RAD, through the distribution of the score rather than through an $\arg\max$.

Our own two arms are the other way. Their greedy loop takes $\arg\max_j d_j$ over the whole window, deflates the chosen direction out of every remaining row, and repeats: twenty best-of-$V$ picks a round, against PINNACLE's and BADGE's one. They are still not top-$B$ rules, because the score they maximise is not the score they started the round with, but $V$ reaches them through an $\arg\max$ at every pick and through a distribution at none. So the ladder does contain arms the quantile argument applies to most directly, they are ours, and they are the arms that collapse hardest on `poisson1d`. We report the alignment and do not read it as evidence: two arms is not a test of the identity, and the confounds in Appendix K apply to exactly these two.

**And their scores are the same quantity.** PINNACLE's embedding is $E = CV\Lambda^{-1/2}|r|$ over a reference set; BADGE's is $G = R\,r$ with $R$ the Nyström factor, which is $CVW^{-1/2}r$. These are the same operator up to a regulariser and an independent draw: PINNACLE floors the eigenvalues at $10^{-3}\lambda_{\max}$ and may truncate to the top sixteen pairs, the factor discards below $10^{-10}\lambda_{\max}$ with no ridge, and for $V > 64$ the reference and landmark subsets are drawn independently. So $\|E\|^2 = r^2 k_z^\top K_{\text{ref}}^{-1} k_z$ and $\|G\|^2 = r^2\tilde{\Theta}_{zz}$ rank the candidates identically. `scripts/score_identity.py` reads both scores out of the harness's own selectors and compares them on `poisson1d` and `crenel`, at three pool sizes, with PINNACLE's top-16 truncation on and off, at initialisation and after 200 steps, drawing the reference and landmark subsets *independently* as the arms do: Spearman between 0.999995 and 1, and the same $\arg\max$ in all forty-eight rows. The truncation axis is close to a no-op, the two settings giving identical values in 21 of 24 pairs, so those forty-eight rows are twenty-four distinct measurements. Read it as identical to five decimals rather than exactly. What differs is the direction, not the score, and so the selected sets differ: PINNACLE uses $|r|$ and BADGE the signed $r$, the reference and landmark draws are independent, and on PINN tasks only PINNACLE augments its basis with boundary rows, which on `crenel` do not exist. Two published methods, two derivations, one per-point score. **We do not offer their agreeing on the crossing rung as corroboration.**

**What remains of the difference is as large as a difference between rules.** On `crenel` at $V = 800$ the two disagree about whether the run fails on 43 of 64 seeds, and their mean per-seed gap in test $R^2$, 0.369, exceeds PINNACLE's gap to RAD, 0.317, a rule with an unrelated motivation; at $V = 200$ the two gaps are 0.2196 and 0.2197. Their *aggregates* are close and not separable (mean difference 0.075, paired Wilcoxon $p = 0.29$), so a table of means shows two methods agreeing where the seeds show one score realised two ways. The literature being compared here reports a mean over three to ten seeds (§6): at that sample size, what a head-to-head between these two resolves is which reference set was drawn.

And the disagreement is itself a function of $V$. Across the ladder the two differ on 0, 2, 20, 43 and 32 of the 64 seeds: at $V/B = 1.25$ a shared score gives a shared verdict on every seed, and by $V/B = 20$ it gives a different one on two thirds of them. The quantity Section 4 shows moving the outcome also governs how far one score's two implementations drift apart, which is the argument of this paper arriving where we did not put it. `scripts/samescore.py` prints the table.

**The rule in our own opening is the one that fits, and it is now on the ladder.** Residual-based refinement adds the $m$ largest residuals in its sampled set: a top-$B$ of $V$, exactly the object the identity describes. For every version of this paper before this one it was absent from the axis,

which was the largest hole in it; it is there now, as RAR-G, and it is the most exposed arm we have measured (Appendix L). dynamicAL (Wang et al., 2022a), which takes $\mathrm{argsort}(-d)[:B]$ and is already windowable, is a second such rule and is run on both tasks at all five rungs (Appendix L).

**It has been run at one end of it.** Seven tasks, $B = 40$, 4000 steps, re-selection every 100, five seeds, scoring the whole pool, so $V/B = 7.5$ on the three $N = 300$ tasks and 30 on the four at $N = 1{,}200$:

| task | $V/B$ | dynamicAL | uniform |
|---|---|---|---|
| `crenel` | 7.5 | 5/5 | 0/5 |
| `sinjump` | 7.5 | 3/5 | 0/5 |
| `blobs_boundary` | 7.5 | 0/5 | 0/5 |
| `poisson1d` | 30 | 5/5 | 3/5 |
| `nlrd1d` | 30 | 5/5 | 1/5 |
| `vburgers1d` | 30 | 5/5 | 0/5 |
| `burgers1d:0.05` | 30 | 1/5 | 0/5 |

It fails 24 of 35 against the control's 4, and the paired mean is negative on all seven, from $-0.021$ to $-5.603$.

**It does not test the identity.** Three things stop it. The identity is a statement about what changes *as $V$ grows*, and this is one rung per task: a level, not a slope. It is catastrophic at the small end too, failing 5/5 on `crenel` at $V/B = 7.5$ where PINNACLE on the ladder fails $18/64$ at $V/B = 5$, so whatever is wrong with it is not something a larger candidate set brought on. And the failures are divergence rather than a poorly chosen forty: its median $R^2$ is $-5.79$ on `nlrd1d` and its mean $-7.90$ on `poisson1d`, where a network predicting the mean would score 0.

**The difference the head-to-head table turns on, stated plainly.** Our whole-pool PINNACLE arm scores all $N$ candidates. PINNACLE as published draws a fresh candidate pool "around $4\times$ as large as the selected points budget in each round" (Lau et al., 2024). The whole-pool arm is therefore an *ablation* of PINNACLE with its own subsampling removed; the faithful implementation is the windowed arm, whose $V = 200$ is slightly larger than PINNACLE's $4B = 160$. We label the two accordingly throughout and do not attribute the whole-pool arm's failure rate to the method.

## Q THE FAILURE THRESHOLD IN THE UNITS THE FIELD REPORTS

**One signed-rank implementation, named.** Every paired Wilcoxon in this paper comes from `scripts/pairedstats.py`, which is a two-sided normal approximation with a tie correction and is the only signed-rank test in the codebase; it exists because two scripts once had different ones. On the significant cells of Appendix X's within-rule column the exact test would give a $p$ four to sixteen times *smaller*, so the approximation is the conservative choice, and no rounded value printed anywhere in this paper moves by more than one digit in its last place under it.

Every count here is a re-reading of `runs/logDF_crenel_V*` at a different cut of `test_acc`; no run is specific to this appendix.

**Fisher against McNemar, since the arms share seeds.** Every count here is compared with Fisher's exact test, which treats the arms as independent samples. They are not: an arm and its control run on the same 64 seeds, so the paired test is McNemar's. On the three `crenel` crossings the discordant pairs are $(16, 0)$, $(9, 1)$ and $(10, 1)$, giving McNemar $3.1 \times 10^{-5}$, 0.021 and 0.012 against Fisher's $1.3 \times 10^{-4}$, 0.030 and 0.016, so Fisher is conservative *in these three cells*. It is not in general: the controls here fail 0, 1 or 2 times, and on a near-empty margin the exact McNemar test discards the concordant pairs and discretises, so across every panel it is the weaker test in most cells where an arm fails more than its control. Two crossings move as a result and we name them rather than take the favourable test: on `burgers1d` against the re-drawn control RAD's first crossing is $V/B = 30$ under Fisher (20 gives 0.053) and 20 under McNemar (0.041); on `poisson1d` PINNACLE at 20 is 0.062 against 0.031. One moves the other way: `RAD+cov` at $V/B = 2.5$ on `burgers1d` against the held control is 0.040 under Fisher and 0.064 under McNemar, so its "never crosses" holds under the paired test and not under the one we report. The three cells below are stronger under the correct test; we keep Fisher because it is what this literature reports. It does not change the

multiple-comparison picture of Section 4: after Holm over the same thirty, McNemar leaves BADGE at $0.52$ and RAD at $0.29$, which is why the paired effect-size reading is primary either way.

**How much the crossing rung depends on the threshold.** The threshold is a choice and the crossing is defined against it, so here is the sweep. On `crenel`, the first $V/B$ at which a rule is worse than the *held* uniform draw with Fisher $p < 0.05$:

| threshold | uniform, first rung | PINNACLE | BADGE | RAD |
|---|---|---|---|---|
| $R^2 < 0.3$ | $1/64$ | 20 | 20 | never |
| $R^2 < 0.5$ (used here) | $2/64$ | 5 | 5 | 20 |
| $R^2 < 0.7$ | $11/64$ | 5 | 5 | 20 |
| $R^2 < 0.9$ | $56/64$ | 20 | 20 | never |

**The crossing rung is stable across $[0.5, 0.7]$ and moves outward at both ends**, and RAD's crossing exists only inside that band. At $0.9$ the control fails on $56$ of $64$ and the cell is close to void by the rule of Section 2; at $0.3$ almost nothing fails and there is little to detect. So the honest statement is not that $0.5$ is conservative, which is a separate and true point about the field's own cut, but that the crossing is a two-rung-wide result inside a two-threshold-wide band, and outside it the ordering survives while the rung does not.

$R^2 = 1 - \mathrm{MSE}/\mathrm{Var}(u^*)$ and relative $L^2$ error is $\sqrt{\mathrm{MSE}/\mathbb{E}[u^{*2}]}$, so for a given task the two differ only by the constant $\mathrm{Var}(u^*)/\mathbb{E}[u^{*2}]$, computed on the test set:

| task | rel. $L^2$ at $R^2 = 0.5$ | at $R^2 = 0$ |
|---|---|---|
| `poisson1d`, `nlrd1d`, `burgers1d:0.05` | 0.71 | 1.00 |
| `vburgers1d` | 0.56 | 0.80 |
| `sinjump` | 0.54 | 0.76 |
| `crenel` | 0.50 | 0.70 |

A run counted as a failure is therefore wrong by 50–71% in relative $L^2$. Rohrhofer et al. (2023) classify a PINN run as unsuccessful above $15\%$. Our threshold is conservative by a factor of three or more, and every failure rate in this paper would rise, not fall, under the stricter one.

## R THE HEAD-TO-HEAD WITH PAIRED INTERVALS

From `runs/logDB_{task}_s100`. Table 1 reports failure rates only, because five tasks with an interval column do not fit a page and the rate is the lead statistic. This is the same data with the paired mean difference against random selection and its $95\%$ CI beside each rate.

**The table is not read at equal $V$, and the caption says so.** The five published arms score all $N = 1{,}200$ candidates; our two window internally at $V = 200$, which is the `--visible` flag in `runs/logDB_{task}_s100`'s recorded `argv`. The size of it is not small and is the paper's own axis: on the $64$ seeds the two families share, our arms fail $9/64$ and $8/64$ at $V = 200$ and $24/64$ and $26/64$ at $V = N$ (`runs/logDF_poisson1d_V1200`), while the held control is bit-identical across the two, $35/64$ in both, because it never reads $V$ either. Comparing the $15/100$ and $12/100$ printed above with the 64-seed figures would mix denominators; like for like the gap is wider, not narrower. Section 2 now lists this as the third exception to "two arms differ only in which points are chosen".

**And here is the table at one $V$.** The ladder's $V = 200$ rung *is* the equal-$V$ cell, on the three of these five tasks that have a ladder, at $64$ shared seeds and the same budget, network and step count. `nlrd1d` and `vburgers1d` have no ladder and are missing from it, which is the limit of the table rather than a choice about which tasks to show.

**It costs us the `poisson1d` column.** At $V = N$ our two arms fail $15/100$ and $12/100$ there against RAD's $16/100$, PINNACLE's $14/100$ and BADGE's $20/100$, which reads as a tie among the pointwise rules. At $V = 200$ the published pointwise rules improve to $5/64$, $5/64$ and $8/64$ while ours barely move, $9/64$ and $8/64$, so **ours become the worst two of the six rules on that**

Table 17: Table 1's arms at ONE candidate set, $V = 200$, $B = 40$, $N = 1{,}200$, 64 shared seeds a cell: the count of seeds whose test $R^2$ ends below 0.5. The published arms carry `win:` and score a fresh 200-candidate window at each re-selection, which is what our two do internally, so every row here inspects the same number of candidates per point kept. The control is the held draw. From `runs/logDF_{crenel,poisson1d}_V200` and `runs/logEH_sinjump_V200` by `scripts/equalv_table.py`.

| arm | `crenel` | `poisson1d` | `sinjump` |
|---|---|---|---|
| random (held) | 2/64 | 35/64 | 1/64 |
| RAD | 2/64 | 5/64 | 0/64 |
| PINNACLE | 18/64 | 5/64 | 0/64 |
| BADGE | 10/64 | 8/64 | 0/64 |
| GRAD-MATCH | 0/64 | 0/64 | 0/64 |
| ours (addpiv) | 0/64 | 9/64 | 0/64 |
| ours (kepiv) | 0/64 | 8/64 | 0/64 |

**task**, `addpiv` outright and `kepiv` level with BADGE. On `crenel` and `sinjump` equalising separates nobody who was not separated before: every arm fails 0/64 except PINNACLE at 18/64 and BADGE at 10/64 on `crenel`, where RAD matches the control at 2/64, and those two are the arms this paper's own thesis predicts a large $V$ hurts.

This re-analysis was chosen after seeing the runs it re-reads, like the grid of Appendix J, and needs no run of its own: `scripts/equalv_table.py` reads `logDF_{crenel,poisson1d}_V200` and `logEH_sinjump_V200`.

**GRAD-MATCH is at or below both of our arms in every one of the five columns**, 0/100 against 1/100 on `crenel`, 1/100 against 15/100 and 12/100 on `poisson1d`, and level at zero on the other three. Section 4 says the same thing about the $V$ axis from $V/B = 2.5$ upward and not at the lowest rung, where GRAD-MATCH fails 1/64 on `crenel` against 0/64 for both of ours, and Section 5 explains why, but the head-to-head itself only names the arm that beats us on cost. On the failure rate it beats us; on the paired accuracy column of this same table it does not, being below `kepiv` on `crenel` and `sinjump` and decisively above both of ours only on `poisson1d`. It costs 28.9 GFLOP against our 5.2.

Generated by `scripts/main_table.py`, which writes this table and Table 1 in one call, from `runs/logDB_{task}_s100`.

Table 18: **Table 1 with the paired accuracy beside the failure count.** Failures at $R^2 < 0.5$ of 100 seeds per cell and the paired mean against the held uniform control, at one $V$ per task. Generated by `scripts/main_table.py`, which writes this table and Table 1 in one call, from `runs/logDB_{task}_s100`.

| | `crenel` | | `nlrd1d` | | `poisson1d` | | `sinjump` | | `vburgers1d` | | sel |
|---|---|---|---|---|---|---|---|---|---|---|---|
| method | crash | Δ vs rand | crash | Δ vs rand | crash | Δ vs rand | crash | Δ vs rand | crash | Δ vs rand | GFLOP |
| random | 3/100 | — | 9/100 | — | 52/100 | — | 1/100 | — | 3/100 | — | 0.0 |
| re-drawn | 0/64 | – | – | – | 1/64 | – | – | – | – | – | 0.0 |
| RAD | 9/100 | $-0.069 \pm 0.042$ | 2/100 | $+0.128 \pm 0.065$ | 16/100 | $+2.349 \pm 2.032$ | 0/100 | $+0.090 \pm 0.026$ | 1/100 | $+0.051 \pm 0.049$ | 1.2 |
| RAD+cov (ours) | 1/100 | $+0.016 \pm 0.037$ | 2/100 | $+0.112 \pm 0.084$ | 6/100 | $+3.904 \pm 1.730$ | 0/100 | $+0.100 \pm 0.025$ | 0/100 | $+0.058 \pm 0.039$ | 1.2 |
| PINNACLE, ours abl. | 46/100 | $-0.341 \pm 0.058$ | 3/100 | $+0.072 \pm 0.101$ | 14/100 | $+3.038 \pm 1.497$ | 0/100 | $+0.086 \pm 0.025$ | 0/100 | $+0.074 \pm 0.037$ | 29.6 |
| BADGE | 36/100 | $-0.209 \pm 0.052$ | 4/100 | $+0.086 \pm 0.077$ | 20/100 | $+2.986 \pm 1.871$ | 0/100 | $+0.096 \pm 0.026$ | 5/100 | $-0.006 \pm 0.065$ | 28.9 |
| GRAD-MATCH | 0/100 | $+0.154 \pm 0.031$ | 0/100 | $+0.154 \pm 0.058$ | 1/100 | $+3.992 \pm 1.711$ | 0/100 | $+0.149 \pm 0.023$ | 0/100 | $+0.088 \pm 0.037$ | 28.9 |
| ours (addpiv) | 1/100 | $+0.151 \pm 0.034$ | 0/100 | $+0.154 \pm 0.058$ | 15/100 | $+1.867 \pm 2.252$ | 0/100 | $+0.150 \pm 0.023$ | 0/100 | $+0.083 \pm 0.036$ | 5.2 |
| ours (kepiv) | 0/100 | $+0.160 \pm 0.029$ | 0/100 | $+0.154 \pm 0.058$ | 12/100 | $+2.712 \pm 2.073$ | 0/100 | $+0.151 \pm 0.023$ | 0/100 | $+0.088 \pm 0.037$ | 5.2 |

**The same head-to-head against the re-drawn control.** A cold reader given only the results page asked for this and it did not exist: every paired difference above is against the *held* draw, which is the bar Contribution 3 argues is the wrong one. It needs no new training. The 100-seed arm runs and the 64-seed ladder runs share their seeds bit for bit, which `scripts/paired_redrawn.py` verifies before it reports and refuses on: for all 64 common seeds the held control's test accuracy is identical in both files, so this is a pairing and not two populations laid side by side. Arms are from `runs/logDB_{task}_s100` restricted to those seeds, the control from `runs/logET_{task}_V1200`; the setting is Table 1's. Which rung that is does not matter for

the control: `random_resample_exam` is bit-identical at every $V$ we ran, $0/64$ on `crenel` and $1/64$ on `poisson1d` at all five, because it never reads $V$. Holm is over the fourteen tests.

Table 19: Paired difference of each arm against the *re-drawn* uniform control, on the 64 seeds `runs/logDB_{task}_s100` and `runs/logET_{task}_V1200` share, at Table 1's setting. Bold is **Holm-corrected** $p < 0.05$ over the fourteen tests, which is not the convention of Table 16, where bold is an uncorrected two-sided Wilcoxon. Positive is better than the control. The 64 are seeds 0–63, a subset of Table 1's 100, which is what makes this a paired test at all: `scripts/paired_redrawn.py` refuses to run unless the *held* control agrees to $10^{-12}$ on every shared seed across the two run families, and it does on all 64. Generated by `scripts/paired_redrawn.py`.

| task | arm | fail | paired mean | median | Holm $p$ |
|---|---|---|---|---|---|
| `crenel` | RAD | $5/64$ | $\mathbf{-0.097}$ | $-0.086$ | $0.0024$ |
| | RAD+cov (ours) | $1/64$ | $-0.023$ | $-0.027$ | $0.6039$ |
| | PINNACLE, ours abl. | $28/64$ | $\mathbf{-0.357}$ | $-0.339$ | $0.0000$ |
| | BADGE | $25/64$ | $\mathbf{-0.257}$ | $-0.254$ | $0.0000$ |
| | GRAD-MATCH | $0/64$ | $\mathbf{+0.120}$ | $+0.111$ | $0.0000$ |
| | ours (addpiv) | $0/64$ | $\mathbf{+0.113}$ | $+0.106$ | $0.0000$ |
| | ours (kepiv) | $0/64$ | $\mathbf{+0.122}$ | $+0.098$ | $0.0000$ |
| `poisson1d` | RAD | $11/64$ | $-1.244$ | $-0.002$ | $0.4582$ |
| | RAD+cov (ours) | $4/64$ | $-0.011$ | $-0.000$ | $1.0000$ |
| | PINNACLE, ours abl. | $9/64$ | $-0.720$ | $-0.001$ | $0.6039$ |
| | BADGE | $8/64$ | $-0.816$ | $-0.001$ | $0.1880$ |
| | GRAD-MATCH | $1/64$ | $\mathbf{+0.100}$ | $+0.007$ | $0.0000$ |
| | ours (addpiv) | $9/64$ | $-1.363$ | $+0.000$ | $1.0000$ |
| | ours (kepiv) | $8/64$ | $-0.723$ | $+0.000$ | $1.0000$ |
| `crenel` | re-drawn control | $0/64$ | — | — | — |
| `poisson1d` | re-drawn control | $1/64$ | — | — | — |

Three things in it, and only the first is what we predicted. **Two of the three softened pointwise rules lose to the re-drawn control on `crenel` and lose heavily**: our PINNACLE ablation at $-0.357$ and BADGE at $-0.257$, both under $10^{-4}$ after correction, with RAD at $-0.097$ behind them. That is the crossing this paper is about, measured against the control it argues for. It is not the first such statement: Appendix O's ladder already pairs against the re-drawn control, on `crenel` alone. What is new here is `poisson1d`, which that table does not carry, and a second run family for `crenel`. The two agree without being the same data: at the top rung the ladder reads RAD $-0.099$, PINNACLE $-0.350$, BADGE $-0.218$, GRAD-MATCH $+0.137$ and our arms $+0.118$ and $+0.140$, against $-0.097$, $-0.357$, $-0.257$, $+0.120$, $+0.113$ and $+0.122$ here, every sign the same and every magnitude within a fifth. The seventh arm is the exception and is named rather than dropped: RAD+cov reads $-0.048$ there against $-0.023$ here, a factor of two, on the one arm whose difference is small enough for a factor to mean little. The arms come from different runs in the two tables, so that agreement is a replication rather than an identity.

**On `poisson1d` no arm is significantly *worse* than the re-drawn control on this statistic, at this rung, with the family drawn over both tasks.** All three qualifiers are load bearing: Appendix L reports both of our arms crossing this same control on this same task by a Holm-corrected Fisher test at $V/B = 20$, at $7.1 \times 10^{-7}$ and $0.0031$, and a Fisher test over these fourteen cells puts RAD at $0.051$, so the sentence survives by a thousandth: RAD is the arm that comes closest to refuting it, and a family drawn per task instead of over both puts RAD at $0.030$, which refutes it. Every arm whose paired mean is negative there fails Holm: the means are large, $-1.36$ for our own `addpiv` arm and $-1.24$ for RAD, while their medians sit between $-0.002$ and $+0.000$ and the smallest of their corrected $p$ is $0.19$, which is BADGE and is thin enough that the exact test moves it to $0.18$. GRAD-MATCH is the exception in the other direction and is covered below. That is the bimodality of Appendix S doing exactly what it does to a rank test, and it means the `poisson1d` column of Table 1 cannot carry a claim that the arms are beaten by this control, only that they fail more often than it does. Section 3 says the weaker thing.

**And the arms that beat the re-drawn control include ours.** GRAD-MATCH is above it on both tasks, $+0.120$ and $+0.100$, and on `crenel` so are both of our arms, $+0.113$ and $+0.122$, all four under $10^{-4}$.

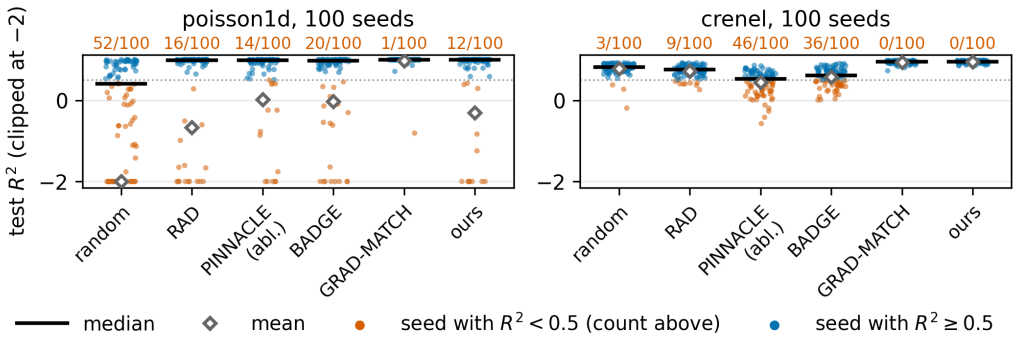



Figure 4: Every seed rather than a summary of them: one dot per run, vermillion below $R^2 = 0.5$ and blue above, axis clipped at $-2$ where failures reach $-30$. On `poisson1d` the outcome is two-valued, so the mean (grey diamond) sits far below the median (black bar) (random's mean is $-3.016$ against a median of $+0.402$) while on `crenel` it is graded and the two nearly agree. That is why the failure rate leads and does not replace. Along the top: BADGE fails $36/100$ on `crenel` where uniform selection fails $3/100$.

## S How far the bimodality goes

Computed by `scripts/bimodality.py`, with the scope of the six sweeps set by `scripts/bimodal_scope.py`. Over the 163 (run, method) cells of the six sweeps this criterion was run on, 10 are bimodal by a two-component criterion (6%), 6 have no seed between the modes, and exactly one has mean and median differing by more than 0.5. That last count is of those six sweeps and not of the paper: `poisson1d`'s uniform row in Table 1 has mean $-3.016$ against median $0.402$, and the head-to-head was not in scope when this was counted. What the count does support is narrower and is what Section 2 claims: the cells that are bimodal are concentrated on the hard tasks, which are exactly the tasks methods are compared on, and Figure 4 shows what that looks like for one of them and, for contrast, for a task where it does not happen.

## T The pool-size null in full

The whole-pool comparison Section 4 draws on for $V/B = 30$ against 500 is $28/128$ against $26/128$: sample odds ratio $0.910$ with a logit 95% interval of $[0.499, 1.660]$, beside the conditional pair quoted above, a null that cannot exclude the odds halving or rising by two thirds. That is why the text calls it a null and not saturation, and why nothing in this paper is measured above $V/B = 30$ on the windowed ladder.

From `logCY_crenel` and `logCX_burgers1d005` at $N = 1{,}200$ and $N = 20{,}000$, 32 seeds each; every count and both Fisher tests below are reproduced by `pool_verdict.py` with `--task crenel --tag logCY`. Holding $B$, $c$, the architecture and the step count fixed and raising $N$ from 1,200 to 20,000 at 32 seeds, the whole-pool rules fail $28/128$ and $26/128$ on `crenel` (Fisher $p = 0.88$) and $3/128$ against $1/128$ on `burgers1d:0.05`, while a windowed control, which draws $V$ exchangeable candidates at every $N$ and must not move, does not: $1/96$ at both pool sizes on `crenel`, $0/96$ at both on `burgers1d:0.05`. The arm has 80% power against a 60% reduction in the failure rate and 62% against a halving, and the interval's upper end permits a 74% increase in the odds, and the conditional odds ratio is $0.911$, 95% CI $[0.476, 1.736]$. The reading rule and the power calculation were fixed before either arm returned, because `burgers1d:0.05`'s base rate cannot reach significance even under complete elimination, so `crenel` was designated in advance as the arm that decides. Pool-size robustness is named as future work by Lau et al. (2024); swept to $N/B = 500$ on the task with the most failure available to remove, it does not appear.

## U THE CONCENTRATION MEASUREMENT OFF THE PDE

**The mediator itself, printed.** Section 5 names concentration and spread as the measured mediator and quotes four numbers for them. Here is the table. Both tasks, at step 50, 8 seeds, from `runs/logDM2_{task}`, by `scripts/mediator_table.py`, which also checks the ordering claim rather than leaving it to the caption: on both tasks the four rules in decreasing concentration are the four rules in increasing spread, in the same order. The margin is thin where the two pointwise rules meet, 0.001 between BADGE and PINNACLE on `crenel` and 0.0008 in spread between the same two on `poisson1d`, so the script prints the smallest gap beside the verdict and the table prints four decimals rather than three, at which those two would read as a tie. The concentration column in the MNIST table later in this appendix is a different measurement, the mean over the six recorded checkpoints over 6 seeds from `logDP`, which is why PINNACLE reads 0.851 here and 0.824 there; step 500 alone would be 0.892.

Table 20: **Concentration and spread, the two measured correlates.** Concentration is the mean quantile of $|e|$ over the selected points and spread their mean nearest-neighbour distance, each normalised so a uniform draw gives 0.5 and 1. Four decimals because the smallest gap between two rules is 0.0008. The MNIST concentration column later in this appendix is a different measurement over different checkpoints. Generated by `scripts/mediator_table.py`.

| | | concentration | | spread | |
|---|---|---|---|---|---|
| task | rule | $V = 50$ | $V = 1200$ | $V = 50$ | $V = 1200$ |
| `poisson1d` | RAD | 0.5344 | 0.6363 | 0.9970 | 0.9753 |
| | PINNACLE | 0.5463 | 0.8507 | 0.9882 | 0.6125 |
| | BADGE | 0.5439 | 0.8484 | 0.9886 | 0.6133 |
| | GRAD-MATCH | 0.5183 | 0.5759 | 1.0109 | 1.0492 |
| `crenel` | RAD | 0.5183 | 0.5458 | 0.9938 | 0.9755 |
| | PINNACLE | 0.4990 | 0.5076 | 1.0098 | 1.0250 |
| | BADGE | 0.4985 | 0.5086 | 1.0098 | 1.0239 |
| | GRAD-MATCH | 0.4715 | 0.4557 | 1.0110 | 1.0499 |

The two the main text quotes are PINNACLE's concentration, $0.546 \to 0.851$, and GRAD-MATCH's spread, $1.011 \to 1.049$: the rule that concentrates hardest is the one whose coverage collapses, and the set-level rule spreads *further* as $V$ grows. On `crenel` the same ordering holds over a much narrower range, 0.456 to 0.546 in concentration, which is the quantitative sense in which that task's mechanism is weaker than `poisson1d`'s.

**The three quantities, defined.** All are measured on identical weights, so that $V$ and the rule are the only things that vary, and all are averaged over window draws and seeds.

- **Concentration** is the mean over $j \in S$ of the quantile of $|e_j|$ within the pool's $|e|$: $\frac{1}{B}\sum_{j\in S}\frac{1}{N}\,|\{i : |e_i| \le |e_j|\}|$. A uniform draw gives 0.5 in expectation and 1 means every selected point sits at the largest residual in the pool. It is universal across rules and is RAD's own criterion, so on that rule it doubles as its own-score quantile.
- **Spread** (dispersion) is the mean nearest-neighbour distance inside $S$ along the domain coordinate, divided by the mean of the same statistic over uniform draws of size $B$, so 1 is a uniform draw and smaller is more clustered. On `mnist` the domain coordinate does not exist and a $k$-nearest-neighbour form in feature space is used instead; every one-dimensional number reproduces under either.
- **Coverage** is the fraction of 10 equal-width bins of the domain that receive at least one selected point, and **coverage drift** is that fraction at $V = 1200$ minus its value at $V = 50$. A drift of $-0.463$ means the selection stops reaching 46 percentage points' worth of the domain as the candidate set grows.

**And the account switches on and off inside a published formula.** RAD's floor $c$ caps concentration and is a parameter its authors set: sweeping it reproduces the whole observed range of coverage drift, $-0.463$ at $c = 0$ against $-0.040$ at the default $c = 1$. The knob is theirs. On `poisson1d` the windowed RAD arm goes $5/64 \to 10/64$ across the $V$ ladder at $c = 1$ and $13/64 \to 21/64$ at $c = 0$:

worse at both ends and steeper in $V$. On `crenel` it does not sweep the range at all, it destroys the run, $62/64$ at $V = 200$ and $64/64$ at $V = 1200$ against $2/64$ and $6/64$ at $c = 1$. So $c = 0$ is not a milder version of a large $V$; it is a rule with no floor, and on the task where the residual is irreducible that is fatal at any $V$. The claim this supports is that the floor is load-bearing and that its author put it there, not that one knob substitutes for the other.

Section 5 measures concentration on two PDE tasks. This repeats it on `mnist:3v5`, binary image classification with labels and no differential operator. **It is not one of the eleven tasks of Section 2**; it is added here because the claim being tested is about what happens away from a PDE residual. Its only other appearance is among the ten tasks pooled in the ablation of Section 5, where it produces no failures under any configuration. It is run in one instrument with the other two so the arms are comparable: 6 seeds, $N = 1200$, $B = 40$, 12 window draws per cell, the same reference trajectory, and the same nearest-neighbour dispersion on all three tasks. The registered prediction was that the concentration a rule gains from a larger candidate set would *not* grow with training here, and it was wrong.

| concentration gain, step $0 \to 500$ | BADGE | PINNACLE | RAD | GRAD-MATCH |
|---|---|---|---|---|
| `poisson1d` | $+0.163 \to +0.336$ | $+0.171 \to +0.334$ | $+0.086 \to +0.123$ | $+0.017 \to +0.060$ |
| `crenel` | $+0.105 \to +0.318$ | $+0.107 \to +0.291$ | $+0.071 \to +0.112$ | $+0.008 \to +0.068$ |
| `mnist:3v5` | $+0.151 \to +0.301$ | $+0.147 \to +0.293$ | $+0.063 \to +0.133$ | $-0.059 \to -0.230$ |
| concentration at $V/B = 30$ | | | | |
| `poisson1d` | 0.823 | 0.824 | 0.651 | 0.556 |
| `crenel` | 0.641 | 0.635 | 0.592 | 0.515 |
| `mnist:3v5` | 0.822 | 0.814 | 0.621 | 0.501 |

**And what it costs there, which is nothing.** The same arms were then run to convergence on the same task, 16 paired seeds, $N = 1200$, $B = 40$, 4000 steps, re-selection every 100: the protocol of Section 3. The outcome on MNIST is not two-valued, so the statistic is a paired accuracy difference rather than a failure count, and that difference is stated rather than smoothed over.

| `mnist:3v5` | $V/B = 2.5$ | $V/B = 30$ | paired difference | Wilcoxon $p$ |
|---|---|---|---|---|
| uniform | 0.8665 | 0.8665 | 0.0000 | — |
| PINNACLE | 0.9010 | 0.8977 | $-0.0033 \pm 0.0130$ | 0.379 |
| BADGE | 0.9059 | 0.9057 | $-0.0001 \pm 0.0129$ | 0.959 |
| GRAD-MATCH | 0.9080 | 0.9228 | $\mathbf{+0.0148 \pm 0.0093}$ | **0.013** |

The uniform row moves by exactly zero on all 16 seeds because it does not read $V$, which is the check that the two runs differ only in the arm. PINNACLE's interval does not exclude a small loss; it excludes anything like the effect on `crenel`, where the same change is $1/64$ failures to $30/64$. And GRAD-MATCH's gain is the other half of this paper's claim measured directly: where the maximised function is the wanted one, more candidates make the greedy solution better, which is what Mirzasoleiman et al. (2015) guarantee for a *monotone submodular* objective, which GRAD-MATCH's is not (Appendix P), and what the rest of this paper only ever observes as an absence of harm.

Eleven of the twelve gain cells grow, each clearing the sum of the two across-seed standard errors. `mnist:3v5` concentrates as hard as `poisson1d` and harder than `crenel`, so the intuition that a label-based score does not localise is not true of this task. The twelfth cell is the result: GRAD-MATCH's gain is negative on MNIST and becomes more negative with training, and its absolute concentration is a uniform draw. Shown twenty-four times more candidates, the one rule whose objective is defined on the selected set selects a *less* concentrated set. Generated by `scripts/pressure.py` and read out by `scripts/conc_trend.py`.

## V Prior art, and what each work established

**Oversearching, quoted in full, because Section 6 only names it.** Quinlan & Cameron-Jones (1995) introduce the term for a theory found by extensive search that is not necessarily overcomplex but

"whose apparent accuracy is also misleading", and report that "exploring larger numbers of potential theories consistently leads to selection of better theories in only one of twelve domains investigated". Their account of why is ours: "for any collection of training data there are 'fluke' theories that fit the data well (according to whatever criterion is employed) but have low predictive accuracy. When a very large number of hypotheses is explored the probability of encountering such a fluke increases." Their beam width runs from 1 to 512, which is a search-width ladder. And the sentence that matters most to Section 5 is their reason for needing a new criterion at all: the choice of width "cannot be made with reference to the $\mathcal{E}$ value alone since this always decreases with further search". That is the divergence between the one-step objective and the trained outcome, stated in 1995, which is why this paper claims the measurement of it and not the observation.

Jensen & Cohen (2000) give the statistics. "A single mechanism is responsible for three pathologies of induction algorithms: attribute selection errors, overfitting, and oversearching. In each pathology, induction algorithms compare multiple items based on scores from an evaluation function and select the item with the maximum score. We call this a multiple comparison procedure (MCP)." Their identity is ours: "if $\Pr(X_i \geq x) = p$, then $\Pr(X_{\max} \geq x) = 1 - (1-p)^n$", with the consequence that the "degree of bias increases with $n$, the number of items examined by an MCP". And the sentence this paper's reporting recommendation rests on: the bias growing in $n$ makes two scores "incommensurable if $n_a \neq n_b$". A rule that scores 100,000 candidates and one that scores 200 are therefore not comparable on their scores, and that is not our observation.

The introduction lists the claims this paper does not make. Nineteen more, kept here for room:

- Selection is a policy whose terminal reward depends on the elapsed step count (Fan et al., 2018).
- Selection pressure should be annealed away late in training (Schaul et al., 2016).
- The very top of a score is the wrong place to select from when selecting aggressively (Paul et al., 2021).
- **The interior optimum at a fixed budget, in a different parameterisation.** Sorscher et al. (2022) write the pruned set size as $\alpha_{\text{prune}} = f\,\alpha_{\text{tot}}$, with $\alpha_{\text{tot}}$ the pool and $f$ the fraction kept, so their $f$ is our $B/V$ and their $\alpha_{\text{prune}}$ is our $B$. Their Figure 1E plots "test accuracy as a function of $f$ and $\alpha_{\text{prune}}$" and its caption reads "for every fixed $\alpha_{\text{prune}}$, there is an optimal $f_{\text{opt}}$", which is our curve under other letters. They also report the reversal at small pool size, "keeping the hardest examples performs worse than random pruning" (and the opposite at large pool size, in the same sentence), and read the whole figure the opposite way round from us, as a prescription to prune harder when the pool is larger. Two things are different in our setting and neither is a defence of the phenomenon: our discarded candidates are free rather than training data thrown away, and our budget is the training budget rather than the dataset.
- **Coverage as a measured mediator.** Zheng et al. (2023) report that importance-based coreset methods, at high pruning rates, "suffer from a catastrophic accuracy drop, performing worse than even random sampling", and give the reason as measurement rather than narrative: their coverage statistic explains "why coresets selected by SOTA methods at high pruning rates perform poorly compared to random sampling because of worse data coverage". The pruning rate is $1 - B/V$. They move $B$ at fixed $V$ where we move $V$ at fixed $B$, which removes the less-data confound but not the mediator, which is theirs.
- A greedy acquisition rule is the one-step truncation of a multi-step optimal policy, which non-myopic Bayesian optimisation states and repairs (Jiang et al., 2020). Wilson et al. (2018) sweep the inner budget of a batch acquisition over $2^{12}$ to $2^{16}$ and report no degradation. It is not a candidate count, it is the work spent maximising an acquisition function by a continuous optimiser, so the quantile identity does not apply to it; the objective is defined and optimised over the batch, not over points.
- The candidate pool is itself a parameter with an interior optimum, stated in an appendix with a figure by Caron et al. (2019), who give the reason we measure it ("the data could be focused in a specific subset of the region of interest"), in a setting that neither of the literatures above cites.
- Restricting a rule's candidate pool is a known efficiency device in deep active learning (Coleman et al., 2022; Tsvigun et al., 2022; Lesci & Vlachos, 2024).
- The parameter exists in exactly one of PINNACLE and PIED, which is what the dividing line predicts. The design side has carried set-level criteria with approximation guarantees since Krause et al. (2008) proved submodularity for sensor placement; the pointwise side has none, and PIED,

which optimises observation inputs by gradient ascent rather than searching discretised ones, has no pool, no quantile and nothing to report.

- Subsampling a candidate set is safe when the objective maximised is the objective wanted, at a cost of $\varepsilon$ (Mirzasoleiman et al., 2015), and aggressive score-based selection can lose to random selection (Ayed & Hayou, 2024; Kirsch et al., 2019). It gives a floor, that a small $V$ is safe, and no sign; and it is about the objective a rule wants, not the proxy it scores. Ayed & Hayou (2024) carry the second in full: they prove and measure that score-based pruning fails at a small retained fraction, that uniform pruning is a strong baseline there, and that mixing uniform draws in helps, which is the same phenomenon in image classification. Their knob shrinks the training set as it sharpens the selection, so the two causes are not separated; ours is the candidate set at a fixed budget.
- **Sweeping a candidate pool at a fixed labelling budget.** SEALS does it: $k$ nearest neighbours swept at a labelling budget of $2{,}000$, five runs per point with a standard deviation, and larger $k$ helping throughout (Coleman et al., 2022). What separates it from Section 4 is not the design but the readout, a mean rather than a failure rate, and the absence of a control that cannot see $k$. AnchorAL's subpool ablation lets the annotated budget vary with the subpool (Lesci & Vlachos, 2024).
- The quantile itself: the best of $V$ draws exceeds the population $1 - B/V$ quantile with probability $1 - (1 - B/V)^V$ whatever the pool size: Jensen & Cohen (2000) for scored candidates, Theorem 6.33 of Schölkopf & Smola (2002) for kernels, applied by Bordes et al. (2005) to one point and a scalar criterion.
- That a set-level loss-decrease objective splits into a modular part and an off-diagonal NTK term (Wang et al., 2022a); and that a PINN learns its NTK's small-eigenvalue directions slowly, so error there survives a finite horizon (Wang et al., 2022b). Those two facts are the whole content of Proposition 1's two halves.
- **The remedy, and both of our conditions, in one paragraph and twelve years early.** Schroff et al. (2015) restrict the $\arg\max$ to a mini-batch because over a whole set "mislabelled and poorly imaged faces would dominate" and the hardest negatives "can result in a collapsed model". That is irreducible error, coverage collapse, and the random window, stated together and never measured. We add the measurement and the ratio at which it turns.
- **An interior optimum in the candidate count itself, in active learning.** Gaillochet et al. (2023) vary the number $Q$ of candidate batches ranked at a fixed acquisition size and report that "applying the biggest pool size does not necessarily yield the best performance. On the contrary, the model performs best when the most uncertain batch is selected from a pool containing $10$ or $100$ different batches. Increasing the pool of choices by $10$ or $100$ does not lead to significant improvements and can lead to worse performances." Their $Q$ is our $V/B$ in batch units, with $Q = 1$ equal to random selection and $Q \to \infty$ to top-$k$, which is the axis of Section 4. Two things are not there and one of them is the whole of what we claim: they draw no per-rule conclusion, and they ran four acquisition functions and reported two. If that unreported pair splits by whether the score removes itself, this was found in $2023$ and not stated.
- **The result and the remedy together, in contrastive learning.** Mitrovic et al. (2020) decouple the number of negatives from the batch size and report that "for a fixed batch size performance actually degrades as the number of negatives is increased", with the intervention being to restrict "the sum in the denominator ... to be over a random subset of all negative examples available in the current batch". Fixed budget, more candidates, worse, fixed by a random window: our headline and our recommendation, in $2020$ and in a different loss. Their account is the gradient's signal-to-noise ratio, not coverage. Ash et al. (2022) then prove "the existence of a collision-coverage trade-off suggesting that the optimal number of negative examples should scale with the number of underlying concepts in the data", which is an interior optimum in the candidate count with coverage as one of its two terms. The structural difference, and it is the only one: their negatives enter a denominator rather than an $\arg\max$, so there is no quantile and no selection rule to sort.
- **The dividing line, in exploration.** Burda et al. (2019) split prediction error into four sources, of which "factor 1" is epistemic, "prediction error is high where few similar examples were seen by the predictor", and state that "factor 1 is what allows one to use prediction error as an exploration bonus". Their factor 2 is aleatoric and their factor 4 is "learning dynamics", "prediction error is high because the optimization process fails to find a predictor in the model class that best

approximates the target function", which is our second condition under another name. Pathak et al. (2017) give the picture: an agent "is unaware that some parts of the state space simply cannot be modeled and thus the agent can fall into an artificial curiosity trap and stall its exploration". RND's whole design is to make the score removable by acting, which is the safe side of our line. What is not there is any connection to how many candidates the $\arg\max$ ranges over.

- **The mechanism itself, in evolutionary computation, thirty years before us.** Keeping the best $B$ of $V$ *is* truncation selection at threshold $T = B/V$. Blickle & Thiele (1996) prove in closed form that its loss of diversity, the proportion of the population not selected in a round, is exactly $1 - T$ (their eq. 4.4), state that "a high loss of diversity increases the risk of premature convergence", and show the loss is "an inherent property of a selection method" rather than of the fitness distribution. Miller & Goldberg (1995) give the consequence in full: "if the selection pressure is too low, the convergence rate will be slow ... If the selection pressure is too high, there is an increased chance of the GA prematurely converging to an incorrect (suboptimal) solution." Our $V/B$ is their $1/T$, our coverage mediator is their loss of diversity, and we add nothing to the mechanism. What is not there is a direction. Their statement is a two-sided risk with an interior optimum, and Blickle & Thiele (1996) hedge it explicitly: "in general it depends on the problem and on the representation of the problem to be solved whether a low loss of diversity is 'advantageous'." Monotone degradation past a threshold, at a fixed training budget, on published neural-network selection rules, does not follow from either. Appendix W.4 answers the sharper form of the same objection.

## W THE SAME AXIS IN FIVE OTHER LITERATURES, READ CLOSELY

**Decoding. The thesis sentence is Vinyals et al.'s, from 2016.** Reporting a captioning system whose best beam width was 3, Vinyals et al. (2017) write: "as the beam size increases, we score more candidate sentences and pick the best according to the obtained likelihood. Hence, if the model was well trained and the likelihood was aligned with human judgement, increasing the beam size should always yield better sentences." That is this paper's premise and its refutation in two sentences, at $B = 1$. They also give a coverage mediator: "by reducing the beam size ... we increase the novelty of generated sentences. Indeed, instead of generating captions which repeat training captions $80\%$ of the time, this gets reduced to $60\%$." Koehn & Knowles (2017) make it the canonical result across eight language pairs and Stahlberg & Byrne (2019) give the cleanest measurement of both halves at once: exact search reduces search errors to zero and BLEU to 2.1, because the model's global optimum is the empty string more than half the time. Ott et al. (2018) ask it in one line, "why does the BLEU accuracy of translations found by larger beams deteriorate rather than improve despite these sequences having higher likelihood?", and answer it with copy noise, which is regressional Goodhart in 2018.

**Decoding again. Ippolito et al. ran this paper's design in 2019.** Ippolito et al. (2019) hold the number of outputs kept at 10 and vary the candidate pool between 10 and 100, then measure diversity of the *kept* set with four statistics. On captioning they report that "the diversity of the captions generated by random sampling actually decreases when oversampling". Fixed keep-count, swept candidate pool, coverage of the kept set as the readout: that is our experiment. Their conclusion is not ours, and we state it in their words: "we can conclude that over-sampling then ranking does not have a sizeable effect, either negative or positive, on quality", and over-sampling improves beam search's diversity even as it hurts sampling's.

**Best-of-$n$. The pressure already has a closed form, and it is not ours.** Best-of-$n$ is $B = 1$ with $V = n$, and Gao et al. (2023) sweep $n$ over three decades and watch the gold reward rise and fall. They do not use a quantile as the $x$-axis; they use $\mathrm{KL}_{\text{bon}} = \log n - (n-1)/n$, from Appendix G.3 of Stiennon et al. (2020). Our $1 - B/V$ is the same idea for general $B$ and we make no claim on it. The mechanism behind both is older still: Smith & Winkler (2006) prove that the gap between the best estimate and its truth, which they call expected disappointment, "increases with the number of alternatives considered".

**The general law is named, and it has a theorem.** Karwowski et al. (2024) state it as "Goodhart's law, which predicts that increasing optimisation of an imperfect proxy beyond some critical point

decreases performance on the true objective", prove a geometric account of it in Markov decision processes, and derive an optimal early-stopping rule. Nothing in this paper adds to that statement. What we add is a way to tell, for one family of rules and before a run, which side of the critical point a given $V$ puts you on.

**Two published sweeps of this axis, and what they say.** Two papers vary the candidate pool at a fixed keep-count and report an outcome, and both report a sign that changes with the dataset. Mindermann et al. (2022) hold $n_b = 32$ and adapt $n_B$, sweeping the kept percentage over $\{5, 10, 15, 20\}$, that is $V/B$ from 20 down to 5. Their Figure 8 gives speedup, epochs to a target accuracy, not final quality: on CIFAR-100 it falls monotonically as the pool grows ($2.9\times$ at $V/B = 5$ to $1.2\times$ at $V/B = 20$), on CINIC-10 it peaks in the middle, and on CIFAR-10 it rises steeply the other way. They read this as "the percentage $n_b/n_B$ of datapoints selected per batch has different effects across datasets" and draw no mechanism from it. Jiang et al. (2019) sweep selectivity and report final test error, which is the right axis, but move the kept budget with it: sharper selection is worse on CIFAR-10 and CIFAR-100 and better on SVHN. Their caption says "higher selectivity gives faster training but can increase error".

RHO-loss is the rule that already subtracts an estimate of the irreducible loss, so whatever degrades it in $V$ on CIFAR-100 is not irreducible error being concentrated. Either their estimate does not remove the part that matters here, or our second condition is not necessary. We have not run their setting and cannot say which, and this is the strongest open objection to the mechanism in Section 5.

**The two results that undo the effect, and why we report them.** Two papers remove the beam-search curse with a one-parameter fix. Murray & Chiang (2018) "show that correcting the brevity problem almost eliminates the beam problem"; Cohen & Beck (2019) "show that constraining beam search to avoid large discrepancies eliminates the performance degradation". Both say the curse is an artifact of a badly shaped criterion rather than a consequence of optimisation pressure as such.

### W.1 The same knob in greedy kernel interpolation

Greedy kernel interpolation has run this experiment, and it comes out the other way, for a reason this paper's second condition names exactly.

$P$-greedy selects the next point at the $\arg\max$ of the power function $P_{X_N}$, a per-point score. It is as pointwise as RAD. Its selected points are nevertheless *asymptotically uniformly distributed* (Santin & Haasdonk, 2017), and the $\gamma$-stabilized family of Wenzel et al. (2021), which restricts the $\arg\max$ to the well-separated part of the domain, has $\gamma = 1$ as $P$-greedy and $\gamma = 0$ as unrestricted greedy, where "all the points are clustered close to" one point. $f$-greedy selects at the $\arg\max$ of the pointwise interpolation residual, which is RAD's criterion written in another notation, and Wenzel et al. (2023) obtain rates that are $L^p$-optimal up to a poly-logarithmic factor for $p \in [1, 2]$ and *faster* for $p > 2$, against the rate an asymptotically uniform scheme attains, *for targets in the kernel's native space*.

The hypothesis is what separates the two settings and it is load bearing in their proofs, every bound carrying a $\|r_{n+1}\|_{\mathcal{H}_k}$ factor that is infinite off the native space. Concretely: an interpolant reproduces $f$ at the points it is given, so selecting a point sets that point's score to zero and the $\arg\max$ must move. Under a finite training horizon it does not: the residual at a selected point decays by $1 - e^{-\lambda t}$ in the eigendirections of $K$, so mass at $\lambda t \lesssim 1$ survives, the score at the selected point survives with it, and the $\arg\max$ stays where it was. **A larger candidate set then buys near-duplicates of a point the model was never going to fix.** That is Proposition 2 read as a statement about the criterion rather than about the error, and it is why the same rule is optimal in one setting and exposed in the other.

Chen et al. (2025) diagnose the unsafe end in the same words we would use: greedy pivoting "is entirely based on exploiting the large diagonal entries without exploring any smaller ones", and "heavily emphasizes outliers", with a temperature family in which greedy is the zero-temperature limit and uniform sampling the infinite-temperature one. What they measure is the trace-norm error $\mathrm{tr}(A - \hat{A})$, a worst case over the native-space ball rather than the error of a particular residual, so their $\Theta(N)$-versus-$r$ separation is a different statement from Proposition 4 and neither implies the other. Their knob is also not ours: theirs is how randomly to sample at a fixed pool, ours is how large the pool is.

One thing this literature shares with the one the paper is about. The candidate sets their $\arg\max$ ranges over are stated and then fixed: $831$ and $3 \times 10^4$ in Wenzel et al. (2021), $2 \times 10^4$ in Wenzel et al. (2023). None of them sweeps that number either.

### W.2 The trust-region reading, in full

A reader may object that a candidate set is not a search width at all but a trust region, chosen small because the score is only locally reliable.

Acquisition functions, coreset scores and pruning metrics all rank candidates by myopic proxies over pools of their own, none of them measured here. Proxy overoptimisation (Gao et al., 2023) and the beam-search curse (Koehn & Knowles, 2017; Stahlberg & Byrne, 2019) are the same shape with the count as the pressure, and we claim the knob rather than the phenomenon.

### W.3 Score staleness over one re-selection interval

Kirsch et al. (2023) find that "freezing scores early on greatly harms performance while doing it later has only a small effect", and our damage is late, so the two asymmetries have to be reconciled rather than both asserted. Over one re-selection interval a rule's score becomes *more* self-consistent between the first and last checkpoint of the interval, on `crenel` from $-0.20$ to $+0.69$. That holds in all twelve task-and-score cells of the *fixed*-interval arm, restricted or not, the restriction never binding there. The read-out's own pre-registered restriction, to windows of at least ten optimiser steps, applies to the *matched* arm, where it is the one to quote and is weaker: eight rise, three are flat and one falls, the fall being a cell that starts at $+1.0000$ and has nowhere to rise. We report the restricted tally because the script that defines the restriction says it is the one the write-up leans on. Unrestricted, that same *matched* arm is six rising, five falling and one flat, so the restriction is what makes the direction legible. The late window is therefore where a score is *least* stale, and the two findings are about different quantities: theirs degrades the information a rule uses, ours removes candidates from it, and an $\arg\max$ rule has failure modes at both ends whose driving quantities move oppositely in $t$.

**The same question, asked of the one paper that raised selection pressure and watched it fail.** Loshchilov & Hutter (2015) sweep what they call selection pressure, a rank-decay temperature rather than a candidate count, and report that at the strongest setting "the cost function stops decaying at later epochs and a divergence may occur", adding that "the use of very strong selection might lead to divergence". Their cure is to recompute the losses, which "drastically improves the results" because it gives "a more consistent ranking of the datapoints". So their high-pressure failure is diagnosed as *stale scores* and removed by refreshing them. Ours cannot be the same thing: every arm here recomputes its criterion at every re-selection, $41$ times per run, and the measurement above says the score is at its most self-consistent exactly where our damage is worst.

### W.4 Selection pressure in evolution strategies, and the ratios that literature recommends

**What they own.** The interior optimum in $B/V$ is a solved problem there. Beyer & Schwefel (2002) derive it twice, on two models, and get the same answer both times: "the optimum truncation ratio becomes $\mu/\lambda \simeq 0.270\ldots$" on the sphere, and the same $0.27$ again on the noisy sphere. That is $V/B \approx 3.7$, just below PINNACLE's default of $4$ and below our measured crossing at $5$. **The qualifiers are theirs and we keep them**: the sentence quoted is about the $(\mu/\mu, \lambda)$-ES with intermediate recombination, in the limit $N \to \infty$ and $\lambda \to \infty$, where it approaches the $(1+1)$-ES optimum. It is not a property of truncation selection in general, and the numerical coincidence with $4$ is a coincidence until someone derives the analogue for a kernel filter, which nobody has. They also state the coverage mechanism: "if the selection pressure is too high and the population size too small, there is a tendency to search in subspaces of the $N$-dimensional search space." CMA-ES does not inherit it: Hansen & Ostermeier (2001) set $\mu = \lfloor \lambda/2 \rfloor$, a ratio of $0.5$, which is the opposite end of the same axis.

**The objection.** Every default in that literature couples the kept count to the candidate count. CMA-ES sets $\mu \approx \lambda/2$ and, when it recommends a larger $\lambda$, says to enlarge it "choosing $\mu$ accordingly"; the cross-entropy method fixes an elite *fraction* $\rho$, typically between $0.01$ and $0.1$ (de Boer et al.,

2005). Holding $B$ fixed while $V$ grows sends the elite fraction below every default that literature recommends, and Hansen & Ostermeier (2001) advise the opposite ratio, "large $\mu$ and if need be a larger ratio of $\mu/\lambda$ up to $0.5$ are preferable".

**What cuts the other way, reported because it does.** Two results in the same literature run against us. Wang & Ba (2020) hold the elite count fixed and sweep the candidate population from $100$ to $2{,}000$, and report that two of their three planners "cannot increase their performance by having bigger population size": flat, not falling. And the cross-entropy method's own adaptive scheme raises the candidate count at a fixed elite count deliberately, as the remedy when progress stalls (de Boer et al., 2005), which is the opposite prescription to ours. Neither measures a coverage statistic, and neither is in a setting where the criterion is a truncated proxy for a terminal error, which is the conjunction Section 5 argues is what decides the sign.

## X THE BUDGET AXIS, AND WHERE THE EFFECT STOPS

Every headline in the main text is at $B = 40$, inside the regime Section 2 locates and then works in. This sweeps $B$ at $N = 20{,}000$ with 32 shared seeds and reads the paired difference against the uniform control, from `runs/logEM_{crenel,poisson1d}_B{80,160}_R{2.5,30}`, where the `R` suffix is $V/B$, because at these budgets uniform stops failing and the failure-rate axis goes void by our own two-sided rule. The band was read at four seeds before any arm: `crenel` at $B = 80$ has $0.063$ of headroom above uniform, $B = 160$ has $0.015$, and `poisson1d` at $B = 160$ has a control at $0.9987$, where nothing can be won and something can still be lost. The registered void check passes too: the $N = 20{,}000$ control matches the $N = 1{,}200$ one, $0/32$ on `crenel` and $2/32$ on `poisson1d` at $B = 80$ in both. At $B = 160$ and $V/B = 30$ on `poisson1d`, the task Section 4 reports as never crossing, two rules fall below the control on the paired mean: GRAD-MATCH by $-2.44$, which is the difference against the control and not the $-2.45$ in the table below, that being the same rule's change across $V$; the two quantities agree to a hundredth here by coincidence (Wilcoxon $p = 2.0 \times 10^{-5}$) and PINNACLE by $-1.42$ ($p = 0.0011$). RAD is down $-0.70$ and does not reach significance ($p = 0.56$). GRAD-MATCH's failure counts there go $1/32 \to 8/32$ at $B = 80$ and $0/32 \to 7/32$ at 160. Section 4 reports the sign on `crenel` and the magnitude on `poisson1d`; at four times the budget that split does not hold.

Accuracy is still bought after the failure rate stops moving: the paired gap to GRAD-MATCH on `crenel` decays $+0.145$, $+0.056$, $+0.035$ and $+0.012$ $[+0.008, +0.016]$ from $B = 40$ to 320, positive at every rung and still separated from zero at the last.

| task | rule | $V/B = 2.5$ | $V/B = 30$ | within-rule $\Delta$ ($p$) |
|---|---|---|---|---|
| `crenel`, $B = 80$ | uniform | $0.905$ $(0/32)$ | $0.905$ $(0/32)$ | — |
| | RAD | $+0.004$ | $-0.030$ | $-0.034$ $(0.059)$ |
| | PINNACLE | $+0.047$ | $\mathbf{-0.043}$ | $\mathbf{-0.090}$ $(0.0001)$ |
| | GRAD-MATCH | $+0.053$ | $+0.067$ | $+0.014$ $(0.004)$ |
| `crenel`, $B = 160$ | RAD | $+0.014$ | $+0.005$ | $-0.009$ $(0.050)$ |
| | PINNACLE | $+0.025$ | $+0.026$ | $+0.000$ $(0.48)$ |
| | GRAD-MATCH | $+0.025$ | $+0.016$ | $-0.009$ $(0.005)$ |
| `poisson1d`, $B = 80$ | uniform | — $(2/32)$ | — $(2/32)$ | — |
| | RAD | — $(0/32)$ | — $(1/32)$ | $-0.02$ $(0.80)$ |
| | PINNACLE | — $(1/32)$ | — $(5/32)$ | $-1.65$ $(0.0001)$ |
| | GRAD-MATCH | — $(1/32)$ | — $(\mathbf{8/32})$ | $-1.37$ $(0.008)$ |
| `poisson1d`, $B = 160$ | uniform | — $(0/32)$ | — $(0/32)$ | — |
| | RAD | — $(0/32)$ | — $(3/32)$ | $-0.70$ $(0.076)$ |
| | PINNACLE | — $(0/32)$ | — $(5/32)$ | $-1.42$ $(0.0001)$ |
| | GRAD-MATCH | — $(0/32)$ | — $(\mathbf{7/32})$ | $-2.45$ $(< 10^{-4})$ |

Paired means of (rule − uniform) in test $R^2$ where the cell is graded, failure counts where it is two-valued, and the within-rule column is the change across $V$ on shared seeds with a two-sided Wilcoxon $p$. On `poisson1d` every median stays above $0.95$ while the means reach $-1.45$, so the failure count carries that task and the paired mean carries `crenel`.

**A note on the tags below.** `P-`$x$ names a numbered prediction in the registration note that a paragraph cites, and those notes ship with the supplementary material so each band and refutation condition can be read where it was written. `R1`, `R3` and `R5` are the three registrations this appendix reads: `notes/omp_rank_ceiling_R3.md` measures the rank ceiling offline, `notes/omp_early_stop_R1.md` registers the ablation that tests it, and `notes/randdrop_counts_R5.md` registers the admixture check on the ablation's control. An *amendment* is a dated addition to a note, committed before the cells it covers ran.

**A candidate mechanism for the inversion, measured, tested and refuted.** GRAD-MATCH's pursuit selects from a Nyström factor of at most $m = 64$ columns and runs *exactly* $B$ steps, with no tolerance and no rank guard (`src/selectors.py`). `scripts/omp_rank_ceiling.py` logs the kept rank and the pursuit's relative residual at every step, on both tasks, three seeds, at initialisation and after a 300-step fit:

| state | $B$ | kept rank | picks after $\|\rho\|/\|\text{target}\| < 10^{-8}$ |
|---|---|---|---|
| after 300 steps | 40 | 40–49 | 0–1 of 40 |
| | 80 | 40–49 | 32–41 of 80 |
| | 160 | 40–49 | 112–121 of 160 |
| at initialisation | 40 | 33–36 | 5–8 of 40 |
| | 80 | 33–36 | 45–48 of 80 |
| | 160 | 33–36 | 125–128 of 160 |

So in the state a run spends almost all of its steps in, the pursuit exhausts its directions between $B = 40$ and $B = 80$, and every pick after that maximises $|a_j^\top \rho|/\|a_j\|$ over round-off. That is where this table reports the inversion.

**Removing the ceiling removes the damage; the control two paragraphs below shows the removal is not what does it.** The test that would decide it was registered in `notes/omp_early_stop_R1.md` before it ran and is the ablation that note specifies: break the pursuit when $\|\rho\|/\|\text{target}\| < 10^{-8}$ and fill the remaining $B - r$ picks uniformly from the same candidate set, with $m$ left at 64 so the test cannot be confounded with a better kernel estimate. Twelve cells, the same 32 seeds as the sweep above, both variants in one run so they share every seed. Paired against the uniform control, on `poisson1d`:

| cell | GRAD-MATCH | with the early stop |
|---|---|---|
| $B = 80$, $V/B = 2.5$ | $+0.092$ | $+0.206$ |
| $B = 80$, $V/B = 30$ | $-1.281$ | $+0.203$ |
| $B = 160$, $V/B = 2.5$ | $+0.003$ | $+0.005$ |
| $B = 160$, $V/B = 30$ | $\mathbf{-2.445}$ | $\mathbf{+0.000}$ |

The registered prediction was that the early stop recovers at least half the unmodified arm's gap at $B = 160$, $V/B = 30$, a band of $-1.222$, and that it is refuted if the two land within $0.25$ of each other. It recovers all of it. On `crenel`, where the sweep above reports no damage at these budgets, the two arms are within $0.009$ of each other in every one of the six cells, and the early stop is the *worse* of the two in four of them, which is the shape expected where there is nothing to repair.

**P-R1b, on the four cells the first amendment added for it, holds on three and is refuted by its own letter on the fourth.** That amendment added $B = 40$ at both rungs on both tasks, because the prediction needs a budget at which the pursuit is not exhausted: at $B = 40$ it has 40–49 directions for 40 picks in the trained state, so the early stop should barely fire and the two arms should be nearly indistinguishable. That is *at* the ceiling and not below it, which this appendix states again below as a limit on the paper's headline budget; the manipulation check is weaker for it, and the measured counts below are what settle how much weaker. Registered as $|\text{mean}| < 0.05$ *and* $p > 0.05$. The paired early-stop-minus-unmodified differences are $+0.010$ ($p = 0.99$) and $+0.141$ ($p = 0.35$) on `poisson1d` at $V/B = 2.5$ and 30, and $+0.008$ ($p = 0.37$) and $-0.004$ ($p = 0.15$) on `crenel`. The $+0.141$ exceeds the band while failing to reach significance, so it is refuted by the letter and a null by the statistic, and we report it as refuted. **At $B = 40$ the early stop fires exactly once per run, at initialisation.** No seed is bit-identical between the two arms in any $B = 40$ cell, and the live counters say why: zero replacements across every one of the $32 \times 40$ selections made during training,

and 6.0 of 40 on average in the single selection made at initialisation, at 34.0 pursuit steps. That is the offline table read by phase rather than pooled: it reports 0–1 of 40 trained and 5–8 at initialisation, at kept ranks of 40–49 and 33–36. Both phases land inside their own range, and the whole $B = 40$ difference is one perturbed selection at step 0 propagating through 4000 steps. It also means P-R1b is not quite the manipulation check its registration intended: at $B = 40$ the arms are one selection apart, not zero.

**And the mechanism is not the one we proposed. The control refutes it.** The early stop replaces the post-collapse picks with uniform draws, and at $B = 160$ that is 114.9 of the 160 picks averaged over a run's forty selections, so the arm is in large part a uniform sampler. Before the $B = 160$ cells existed, the note's second amendment registered the control that separates those two readings: run the pursuit to all $B$ steps, then replace *the same number* of its picks with uniform draws, chosen from all $B$ rather than from the tail. We call that arm `randdrop` throughout, and it is the "same count, random picks" column below. Identical admixture, different picks. That last sentence was true of the code and unmeasured in the runs until we measured it, because each arm derives its own replacement count from its own trained model and the randdrop arm trains on a differently diluted set. Registered as P-R5a in `notes/randdrop_counts_R5.md` and read at 34.77 against 34.13 replacements per selection at $B = 80$ and 114.86 against 113.98 at $B = 160$: 1.9% and 0.8% apart, inside the registered 10%, and the control carries the *smaller* admixture in both, which is the direction the note recorded in advance as the one that would strengthen this refutation rather than confound it. P-R1d predicted that arm would recover less than half of what the early stop recovers, and set refutation at its recovering as much.

| `poisson1d`, $V/B = 30$ | GRAD-MATCH | early stop | same count, random picks |
|---|---|---|---|
| $B = 80$ | $-1.281$ | $+0.203$ | $+0.126$ |
| $B = 160$ | $-2.445$ | $+0.000$ | $-0.067$ |
| *the other four cells the same run carries* | | | |
| `poisson1d` $B = 80$, $V/B = 2.5$ | $+0.092$ | $+0.206$ | $+0.131$ |
| `poisson1d` $B = 160$, $V/B = 2.5$ | $+0.003$ | $+0.005$ | $+0.003$ |
| `crenel` $B = 80$, $V/B = 30$ | $+0.066$ | $+0.066$ | $+0.025$ |
| `crenel` $B = 160$, $V/B = 30$ | $+0.016$ | $+0.024$ | $+0.012$ |

**P-R1d is refuted on the cell it was registered for**, `poisson1d` at $B = 80$, $V/B = 30$, and on the $B = 160$ cell beside it. Replacing picks at random recovers 95% of the gap at $B = 80$ and 97% at $B = 160$. What is specific to the post-collapse picks is the remainder: $+0.077$ at $B = 80$, which is significant at $p = 1 \times 10^{-4}$, and $+0.068$ at $B = 160$, which is not ($p = 0.068$). So the rank ceiling is real, measured, coincides with where the inversion turns, and *is not shown to cause it*. What the ablation shows instead is that **on the cells that carry damage, diluting this rule's selection with uniform draws removes it, and which picks are diluted accounts for at most a twentieth of the gap**, which is the coverage remedy this paper recommends for pointwise rules arriving at the one rule it had called protected.

**On the cells that carry no damage the position does matter, significantly.** The two `crenel` cells above are both above the control to begin with, so there is nothing for dilution to repair, and there the early stop beats `randdrop` by $+0.040$ ($p = 8 \times 10^{-5}$) and $+0.012$ ($p = 9 \times 10^{-5}$) over the same 32 seeds. Dropping the pursuit's tail costs about a thousandth; dropping the same number of picks from anywhere costs ten to forty times that. **So dilution is what removes the damage where there is damage, and where there is none, the tail is the cheapest thing to throw away.** Neither half was registered and both are read off the same six cells.

**The live count agrees with the offline instrument in all six readings.** P-R1c asked that the number of picks the early stop replaces, counted inside the sweep rather than in a separate script, match the offline count above to within 10%. Reading it required adding two integers to the sweep's per-seed output, because the harness had been discarding what the selectors return, and re-running the cells. The two phases are separate readings and the offline table gives each a different range:

| $B$ | phase | live replaced | live pursuit | offline (rank) |
|---|---|---|---|---|
| 40 | at initialisation | 6.0 of 40 | 34.0 | 5–8 (33–36) |
| | after training | 0.0 of 40 | 40.0 | 0–1 (40–49) |
| 80 | at initialisation | 46.1 of 80 | 33.9 | 45–48 (33–36) |
| | after training | 34.8 of 80 | 45.2 | 32–41 (40–49) |
| 160 | at initialisation | 126.3 of 160 | 33.7 | 125–128 (33–36) |
| | after training | 114.9 of 160 | 45.1 | 112–121 (40–49) |

Every live figure is inside its own offline range, and so is every pursuit count against the kept rank beside it. The decisive cell was also re-run from scratch and reproduces at $-2.445$ and $+0.000$ to three decimals, so the logging does not perturb the selection it measures. Read from `runs/logFL2_poisson1d_B{40,80,160}_R30_counts2` by `scripts/r1_readout.py`, which prints each prediction's verdict.

**Four limits on what this measures.** The rank is a property of the search space and says nothing about test error by itself. The ablation's admixture confound is the one its own registration named, and the control above tests it rather than settling it. Half of the rank measurement's registered prediction is refuted: the ceiling is present at $B = 40$ at initialisation rather than switching on above it. And at $B = 40$ in the trained state the pursuit has 40–49 directions for 40 picks, so this paper's headline budget sits *at* the ceiling rather than below it.

**It is specific to the pursuit, and that is a narrower statement than it sounds.** The same script runs $k$-means++ on BADGE's own embedding over the same factor: it returns $B$ distinct points at every budget up to 320, and the squared distance it maximises never becomes round-off, so there is no rank ceiling there. Its marginal gain decays all the same, the farthest remaining point at $B = 160$ sitting within a thousandth of the first pick's distance, which is the pool being covered rather than the rank being spent. Only GRAD-MATCH has *this* ceiling; it is not the only rule with a budget ceiling.

**Ten times the budget, and the effect outlives the dividing line.** The sweep above moves $B$ at two rungs of $V$. What it does not have is a full ladder at a budget the field would recognise, which is the shape of this paper's own headline figure, and `notes/budget_order_up_ladder.md` registers one: `poisson1d` at $B = 400$, ten times ours, $N = 20{,}000$, five rungs, 16 seeds. That note also declares, before the run, that the failure-rate axis is expected to go void here and that the verdict is therefore the paired mean against the *re-drawn* control. It did go void: that control reaches $R^2 = 0.9997$ and fails $0/16$ at every rung. Sixteen seeds is a quarter of what the ladders above use and a stated weakness of this cell rather than a hidden one: the cost of a run at ten times the budget is what bought the axis, and every interval here is correspondingly wide.

Table 21: `poisson1d` at $B = 400$ and $N = 20{,}000$, 16 seeds, five rungs of $V/B$. The first two columns are the paired mean of (rule − re-drawn control) in test $R^2$ at the lowest and highest rung; *within-rule* is the second minus the first, which is the statistic the registration reads, with its two-sided Wilcoxon $p$ over the 16 shared seeds Holm-corrected over the three within-rule tests; *fails* counts seeds below $R^2 = 0.5$ at the highest rung, against a control that fails $0/16$ at every rung. GRAD-MATCH's within-rule mean of $-20.97$ sits against a median of $-0.30$, so the mean is carried by a few diverged seeds and both are quoted. The three intermediate rungs are in `runs/logFH_poisson1d_B400_V{V}` and are not printed; the text's "from $V/B = 5$" is read on the per-rung tests over all five. Generated by `scripts/order_up_table.py`.

| arm | $V/B = 1.25$ | $V/B = 30$ | within-rule | $p$ | fails at 30 |
|---|---|---|---|---|---|
| RAD | $+0.0001$ | $-0.1773$ | $-0.177$ | 0.0063 | $1/16$ |
| PINNACLE | $-0.0000$ | $-1.8802$ | $-1.880$ | 0.00021 | $1/16$ |
| GRAD-MATCH | $-0.0970$ | $-20.9711$ | $-20.874$ | 0.0017 | $8/16$ |

**The axis survives an order of magnitude up in budget; what it is worth to a practitioner there does not.** Both softened pointwise rules still degrade in $V$, by far more than that note's registered threshold of 0.01 in $R^2$, and surviving Holm over the three within-rule tests. That is its P-FH1 and it holds. The registration's own wording for that family, "the 15 tests here (three rules by five rungs)", describes a family of per-rung tests rather than the within-rule test it had just specified; the three within-rule tests are the family used. The prediction asks for one rule and PINNACLE's $p$ is 0.0002,

so it holds on either reading, but RAD's $0.0063$ survives a family of three and not one of fifteen. "Both" is therefore a statement about the smaller family. But at $B = 40$ a larger candidate set makes PINNACLE fail 30 of 64 runs, and at $B = 400$ it costs PINNACLE two thousandths of $R^2$ and one run of sixteen, which a Fisher test against the control cannot separate from none. The sentence a practitioner at this scale should take is that $V$'s effect is still measurable and no longer large enough to act on, and that sentence was registered against us before the run.

**The dividing line does not survive at all, and this is the sharpest evidence against our own sorting.** GRAD-MATCH is not merely unprotected here; it is the worst arm on the ladder by an order of magnitude, failing 8 of 16 seeds at the top rung where the control fails none, at a median of $-0.30$, and significantly below the control from $V/B = 5$. The registered prediction that it would not degrade is refuted. Together with the inversion at $B = 160$ above, the sorting of Contribution 2 is a $B = 40$ statement in the strongest sense available: we ran the cell that would break it, at the scale a reviewer would ask for, and it broke.

**Which variable it is, and one thing this does not settle.** The sweep above moved $B$ and $N$ together, so we ran the cell that separates them: $B = 40$, the protocol's own budget, at $N = 20{,}000$. GRAD-MATCH is immune there, improving $+0.044$ on `crenel` ($p = 0.0001$, $0/32$ failures) and flat on `poisson1d` ($p = 0.81$), while PINNACLE degrades as it does everywhere ($-0.420$, $18/32$ against a control at $1/32$). **So the variable is the budget and not the pool density**, and the registered prediction that it was the density is refuted. We have no mechanism for it: why matching the full-pool gradient with four times the picks should stop protecting a rule is not something these sweeps say, and it is not something the terminal-reward model has been asked. And `crenel` at $B = 160$ has only $0.015$ of headroom, twice `staircase`'s but not comfortable, so the disappearance of the effect there is reported and not leaned on.

## Y IS IT THE QUANTILE, OR IS IT THE CHURN?

From `runs/logEL2_{crenel,poisson1d}_V200`, which carry the `winfix:` arms, `winfix:`$R$ being the published rule $R$ run on one window held for the whole run where `win:`$R$ re-draws its window at every re-selection. Shrinking $V$ lowers the quantile the selection sits at, which is the mechanism this paper derives, and it also forces the selected set to churn between rounds, which is a different one and is exploited on purpose elsewhere: R3's release step, the annealed prioritisation of Schaul et al. (2016), the stochastic acquisition of Kirsch et al. (2023). The experiment as described cannot tell them apart, because `win:`$R$ re-draws its window at every re-selection. So we registered the control before running it and ran it: `winfix:`$R$ draws *one* uniform window of $V$ from the run's own seed and holds it for all 41 selections, changing nothing else.

Paired on the same 64 seeds at $V/B = 5$, `winfix` minus `win` in test $R^2$:

| | `crenel` | | `poisson1d` | |
|---|---|---|---|---|
| rule | $\Delta R^2$ (95% CI) | $p$ | $\Delta R^2$ (95% CI) | $p$ |
| RAD | $+0.002$ $[-0.042, +0.046]$ | 0.86 | $+0.070$ $[-0.022, +0.163]$ | 0.16 |
| PINNACLE | $+0.003$ $[-0.090, +0.095]$ | 0.82 | $-4.37$ $[-12.8, +4.0]$ | 0.57 |
| GRAD-MATCH | $+0.004$ $[-0.009, +0.016]$ | 0.98 | $-0.004$ $[-0.011, +0.003]$ | 0.014 |

Five of the six cells are null. The sixth is $-0.004$ in $R^2$ on a rule sitting above $0.99$ with $0/64$ failures either way, detectable because the sign is consistent, 43 of 64 seeds negative, and not because the magnitude is large: $t$-test $p = 0.26$ on the same cell. Either way it is negligible against the $+0.199$ that the $V$ ladder moves GRAD-MATCH's own paired mean on `poisson1d`, from $V/B = 1.25$ to 30. For scale, take `crenel`, where all three intervals are usable. Moving $V/B$ from 1.25 to 30 changes the paired mean by $-0.120$, $-0.377$ and $+0.097$. So these intervals exclude a churn contribution larger than about a quarter of the $V$ effect, on the rule whose $V$ effect is largest, as an interval statement rather than as a power calculation.

**So it is the quantile and not the churn, at this rung.** Holding the window fixed removes the churn and changes nothing measurable in five of six cells, while moving $V/B$ over the ladder moves the same cells by one to two orders more.

**An earlier run at a different cadence disagrees, and here is why it is not this table.** `logEL_crenel_V200` carries the same arms re-selecting every 500 steps rather than every 100, so it makes nine selections where this table's setting makes forty-one. There GRAD-MATCH is $+0.034$ $[+0.017, +0.050]$, $p = 5 \times 10^{-5}$, which would make one of the six cells non-null. The table above is the run at this paper's own cadence.

Two things this does not settle. A fixed window also removes coverage the run never recovers, so the null says that neither churn nor that removal matters at $V/B = 5$ without separating them; a third arm on a slower re-draw cadence would, and is not run. And on `poisson1d` PINNACLE's interval is $\pm 8.4$, because that task's $R^2$ has heavy negative tails, so only two of the three rules carry usable power there and the `crenel` panel is what this rests on.

## Z THE REPLICATION THAT FAILED, IN FULL

From `runs/logEH_{sinjump,staircase,tri_zone}_V{50,100,200,800,1200}`. Registered before the control was read, with the refutation condition written down: the criterion was that PINNACLE's failure count at $V = 1200$ exceed uniform's at Fisher $p < 0.05$, which the pre-read control put at $\geq 8/64$ on `sinjump` and $\geq 6/64$ on the other two.

**Refuted.** On `sinjump`, `staircase` and `tri_zone`, every rule fails $0/64$ at every $V$. No crossing anywhere.

**What replicates instead**, read with a paired difference because the outcome on these tasks is continuous rather than close to two-valued, which is the same reason `mnist:3v5` is read that way. From $V/B = 2.5$ to 30, paired on 64 shared seeds:

| | `sinjump` | | `tri_zone` | |
|---|---|---|---|---|
| rule | $\Delta R^2$ | $p$ | $\Delta R^2$ | $p$ |
| PINNACLE (pointwise) | $-0.045$ | $< 10^{-4}$ | $-0.016$ | 0.004 |
| BADGE (pointwise) | $-0.035$ | $< 10^{-4}$ | $-0.012$ | 0.053 |
| RAD (pointwise, floor $c$) | $-0.027$ | 0.0002 | $-0.018$ | 0.0004 |
| GRAD-MATCH (set-level) | $+0.006$ | 0.002 | $+0.0004$ | 0.027 |

**Two things we got wrong, both worth stating.** `staircase` is uninformative and we did not see that before running it: uniform reaches $R^2 = 0.993$ there, so the gap between the best rule and the control is 0.007 against 0.156 on `crenel` at $V/B = 2.5$, where `crenel`'s runs 0.145 to 0.189 across the ladder and `staircase`'s is 0.007 at every rung, and nothing can be won or lost. Read anyway it moves the other way, and this is exactly what excluding it excludes: PINNACLE $+0.0014$ and BADGE $+0.0006$ over the same range of $V/B$, both at $p < 10^{-4}$, and RAD $+0.0034$ at $p = 0.40$. The exclusion is a judgement made after the run, on a criterion (usable headroom) that the registration did not contain. A control check that counts failures cannot see this, and ours counted failures. And `tri_zone` was registered as a smooth negative control on the strength of its description; its own function has a jump of 1.2 at $u = 0.600$, larger than either `crenel`'s or `sinjump`'s. So this unit contains no negative control, and whether smoothness gates the effect still rests only on the ladders in Section 5.

**And what the failure says.** Falling below uniform needs a third ingredient beyond the two conditions: an abandoned region that actually needs points, which is the complement half of $R(S, t)$ and is derived in Appendix C. `crenel` has one. `sinjump`'s bulk is a smooth low-frequency sine a $2 \times 64$ network fits from almost nothing, so abandoning it is nearly free, and the rule degrades without ever reaching the control.

## AA THE TEST APPLIED TO RULES WE DID NOT RUN

**Three published families have no candidate set at all, and the axis does not reach them.** This paper's quantity is $V$, the number of candidates *scored* per point kept, and a rule that never scores a finite set of candidates has no $V$ to report and no quantile for it to move.

Tang et al. (2023)'s DAS-PINNs "treat the residual as a probability density function and approximate it with a deep generative model", then draw new points from that model: the points are *generated*, not chosen from a pool, so there is no best-of-$V$ anywhere in it. Visser et al. (2026)'s PACMANN "incrementally moves collocation points toward regions of higher residuals using gradient-based optimization": continuous transport of a fixed set, again with no candidates scored. McClenny & Braga-Neto (2023)'s SA-PINNs do not select at all, the "adaptation weights" being "fully trainable and applied to each training point individually", so the point set is fixed and the *weights* adapt.

So Section 5's account predicts all three are immune to the failure this paper reports, and predicts it without running them, which is what a dividing line read off a definition is for. It also marks the limit of the recommendation: "report $V/B$" is advice to the rules that have a $V$, and for these three the corresponding question is a different one we have not asked. Two of the three give the same reason for avoiding a pool that we give for measuring it, PACMANN opening on residual-based sampling being "reported to scale poorly in terms of computational cost".

**A fixed collocation budget is a named published regime, and its exemplar is a prediction.** Nguyen et al. (2023)'s FBOAL keeps the "number of collocation points ... the same during the training" and relocates them, so a fixed budget is not a construction of ours. Their rule "adds and removes collocation points that yield the largest *and the smallest* PDE residuals on the sub-domains": a residual criterion carrying a coverage device, per sub-domain, which is structurally what protects `RAD+cov` here. By our test it should be less exposed in $V$ than RAD. **We do not run it**, so that is a prediction and not a result, and it is stated here rather than after seeing an answer.

**First, what the test is read on, because our own primary baseline settles it.** PINNACLE's *criterion* is set-level: Lau et al. (2024) define a convergence degree $\alpha(Z) = \sum_i \lambda_i a_i(Z)^2$ that is quadratic in the selected set and therefore carries exactly the off-diagonal structure the test asks about, and they prove a generalisation bound for it. Read on that definition the test puts PINNACLE on the safe side. Its two *published approximations* do not inherit that: one samples with probability proportional to $\hat{\alpha}(z)$ at a single point, the other embeds each point and runs $k$-means++. Both are pointwise, and the measured behaviour follows the approximation, not the criterion. It also says where the damage enters in this case: not in choosing a set-level objective but in approximating it pointwise, which is what separates PINNACLE from GRAD-MATCH, whose matching pursuit optimises its objective over sets.

**And it does not sort the one external list we can test it on.** Ivison et al. (2025) report which of their methods get worse with a larger pool and which improve, at $290\times$ our pool size: PPL, IFD and one embedding arm degrade, while another embedding arm and RDS+ improve. Three of those arms are the *same* algorithm, a cosine similarity between each candidate and the target examples, differing only in which encoder produces the representation: NV-Embed-v2 degrades, GTR-base improves, and the training model's own position-weighted hidden states improve. Our first condition returns the same verdict for all three, because a similarity to a fixed target set is pointwise and is not removed by selecting anything. So the condition does not explain their split, and what does differ among those three is the encoder, about which this paper says nothing. it is a failure of coverage rather than a counterexample: the condition separates the rules *we* run, on five tasks, and does not reach theirs.

**Read this as a prediction and not as a result.** It is the first condition only: does selecting a point remove that point's own score inside the training horizon? A rule that fails it is exposed to $V$ *if* the second condition also holds, that the error the score finds is error the network cannot reduce in that horizon, and nothing below measures that.

| rule, and its score | does selecting remove the score? | ours to run? |
|---|---|---|
| *yes, by construction* | | |
| $P$-greedy, $f$-greedy: power function, interpolation residual | Exactly. An interpolant reproduces its data, so the score at a selected point is zero (Santin & Haasdonk, 2017; Wenzel et al., 2023). | no |
| RND: prediction error against a random target | Yes, and its authors say so: a predictor trained on what is visited has "lower prediction errors on examples similar to those on which they have been trained" (Burda et al., 2019). | no |
| RHO-loss: training loss − held-out loss | Yes over the horizon, no within the round. Training on the point lowers its training loss, so the score falls eventually; the held-out term is fixed before training and the rule assumes no cross-effect, so nothing changes at the moment of the pick. **And the subtraction is not what makes this row a yes**: it removes what its authors call the *irreducible* holdout loss, "the lowest loss that the model can achieve without training on" the point (Mindermann et al., 2022), which is the *second* condition engineered out, not the first. | no |
| GRAD-MATCH, BatchBALD, submodular batch objectives | Yes, within a round: the objective is on the selected set, so each pick changes what the next is worth (Killamsetty et al., 2021; Kirsch et al., 2019; Mirzasoleiman et al., 2015). | the first |
| *no, not inside a finite horizon* | | |
| RAR-G, RAD, R3, residual importance sampling: the PDE residual at a point | No. The residual there decays by $1 - e^{-\lambda t}$, so mass at $\lambda t \lesssim 1$ survives whatever is selected (Lu et al., 2021; Wu et al., 2023; Daw et al., 2023; Nabian et al., 2021). | RAR-G, RAD |
| EL2N, GraNd: error norm, gradient norm at a point | Only the reducible part. Their authors already report that the very top of the score is the wrong place to select from (Paul et al., 2021). | no |
| TD-error prioritised replay | Partly, and the gap is engineered around: stochastic prioritisation is there because greedy prioritisation "can lead to a loss of diversity", and importance weights correct the bias it introduces (Schaul et al., 2016). | no |
| PINNACLE, BADGE: $k$-means++ over residual-scaled embeddings | No. What those embeddings carry is the residual, and training removes only its reducible part. Their $k$-means++ step does drop a chosen point's distance inside the round, but that is the second route below and not this one (Lau et al., 2024; Ash et al., 2020). | yes |

**The selection step removes a point's score inside the round**: GRAD-MATCH's orthogonal matching pursuit deflates the atom it takes, and $k$-means++ zeroes the distance of the point it picks. Three of the four rules we run carry that, and two of those three cross below uniform anyway, so deflation inside the round is not what separates them. What GRAD-MATCH has that BADGE and PINNACLE do not is an objective that is the quantity wanted, a match to the full-data gradient, rather than diversity over a pointwise score. Even that has a budget limit, and Section 7 measures it.

**What the table is for, and the one row that pays for it.** The row worth the appendix is RHO-loss. It is a *pointwise* rule, so the first column would put it with EL2N, and it is not: its score is the training loss with an estimate of the irreducible loss subtracted off, and its authors name that term with the word this paper uses for the same object. A rule that removes the irreducible component from its own score is, in the terms here, a pointwise rule with the second condition engineered out of it. Its reported degradation in $V/B$ on CIFAR-100 therefore runs against the account rather than with it: the conjunction above predicts that a rule with the second condition removed is safe. Appendix W gives the two readings and says we cannot choose between them. We did not run it, we make no

claim about how it behaves in $V$, and it is prior art for the idea that selecting on error a model cannot reduce is the failure worth designing against.

**And the remedy is published too.** A coverage constraint bolted onto a pointwise score is what Zheng et al. (2023) do with stratified sampling across importance scores and what RAD's floor $c$ does inside Wu et al. (2023)'s own density. Our windowing is a third instance and the cheapest of the three to apply to a rule you did not write.

## AB WHAT IS NOT IN THIS STUDY

**Two predictions of Appendix C, registered here and untested.** *First*, we refuted a transfer statistic as a predictor of the outcome: the cross term $e_{S^c}^{\top} K[S^c, S] e_S$ degrades most on the pool that is damaged least. It was read three ways, on a uniform draw, on the selected set across $V$, and as an ordering of rules within one task, and none of the three predicts the damage: every reading puts `poisson1d` and `mnist:3v5` the wrong way round. An abandonment ratio, the median residual over the points a rule has not recently touched against the same over the points it has, is a statistic of the trajectory rather than of the state, and was registered and refuted on all three of its predictions: it is flat on `poisson1d` and *falls* on `crenel`, the opposite of the predicted direction. The model says the load-bearing object carries the filter, $K[S^c, S]\, g_t(K_S)\, e_S$, and that the statistic we measured is its $t \to 0$ limit, which is the one horizon at which the model says nothing is happening. That is a specific and cheap thing to measure and we have not measured it. *Second*, optimal experimental design criteria (expected information gain, $D$-optimality, the mutual-information criterion of the sensor-placement literature) are functions of $K_S$ alone. By (iii) they are set functions and should be immune to candidate-pool size, where the residual heuristics of the same application area should not. It is also a prediction our own data now puts in doubt: at $B \geq 80$ a set function is not immune here (Section 7), so the right version of the test compares set-level criteria *at the budget where set-level stops protecting*. It is worth noting that this literature does not report a candidate-pool size for the same reason: Hemachandra et al. (2025) optimise the design directly rather than ranking a pool, so no quantile exists to move.

**One setting on record disagrees**: Tsvigun et al. (2022) shrink the candidate pool in text classification and report no accuracy gain, with a random window worse than a guided one. We expected a label-based score not to localise the way a PDE residual does, measured it, and it does (Appendix U). Nothing here explains their setting.

**The left arm of the curve is not observed**, because $V \geq B$; and the second path along $V/B$, varying $B$, is unavailable, since reaching small ratios by growing the budget removes the headroom that makes the effect detectable. $V/B$ is the axis because it is the quantile, a derivation, not a validated empirical collapse.

**Dimension.** Six of eleven tasks are PDEs and four are on a two-dimensional $(x, t)$ domain, the input dimension of the Burgers problem of Wu et al. (2023) and the convection problem of Daw et al. (2023), but none is two-dimensional in *space*, and nothing here speaks to the regime where the pool must grow simply to cover the domain. The benchmarks that regime is usually run on are Hao et al. (2024)'s PINNACLE and Takamoto et al. (2022)'s PDEBench, "a benchmark suite of time-dependent simulation tasks based on Partial Differential Equations". Neither is used here, and the honest form of the scale objection is that the sign we report is established on one- and two-dimensional problems at $B = 40$ and is untested on either suite.

**The width test, both halves.** One half of it alone reads as void. At $4 \times 128$ on `poisson1d` uniform selection fails $32/32$ and nothing can be read, which is the void half. On `crenel` uniform fails $2/32$, so that half is readable and we report it: PINNACLE fails $17/32$ windowed and $29/32$ on the whole pool, paired $-0.339$ and $-0.753$ against uniform, so the effect is if anything stronger at the wider network. And GRAD-MATCH fails $7/32$ windowed and $9/32$ whole-pool, paired $-0.079$ ($p = 0.14$, not significant) and $-0.182$ ($p = 0.0003$), so one of the two carries it: a second cell, beside the budget sweep, where the set-level rule is not immune. And our own `kepiv` arm fails $11/32$, paired $-0.202$, worse than either.

**We dropped the tasks a reader would say should have been the tests.** On the first one we went back and ran it, and the exclusion was wrong. `burgers1d` at $\nu = 0.01/\pi$ and `allencahn1d`

carry sharp fronts, and a sharp front is exactly the error the network cannot reduce that Section 5's second condition is about. Dropping them for being optimisation-limited looks like dropping the instances that would have tested the claim, and a referee is entitled to say so.

The defence we offered was a prediction. The abandoned share of Appendix C needs the score to *localise*: the damage is the irreducible mass outside the top-$B/V$ slice, and it is only large when the slice is somewhere in particular. On a task the optimiser cannot fit at all the residual is large everywhere and does not localise, so the score separates candidates weakly, a top-$B$ of it is close to a uniform draw, and the differential against a uniform control goes to zero. On that argument optimisation-limited is not a third ingredient of the damage but a condition that *suppresses* it, and we wrote that the exclusion was predicted rather than convenient, registering the consequence: the same $V$ ladder on `burgers1d` at $\nu = 0.01/\pi$, *where the prediction is that no arm separates from uniform at any rung*. A design note frozen days earlier had registered the opposite and we had not read it, which is the whole of how a paper comes to hold two predictions that cannot both be right.

It is run, in Appendix M, and this prediction is refuted while the earlier one is confirmed: six of seven arms degrade in $V$, five cross below the control, and the one that does neither is the set-level rule. The arms do separate, so the exclusion was convenient. But the configuration does not solve Burgers, and Appendix M reports the two floors that establish it: a time-independent predictor scores $R^2 = 0.929$ on the same grid and no cell of that ladder reaches it. The separation is real and paired; the level is not a solved PDE. We then ran `allencahn1d` too, and it is excluded for a different reason than the one we gave: see below.

**Two of the eleven tasks carry no result, and both are saturated at this budget.** The canonical low-viscosity Burgers ($\nu = 0.01/\pi$) is not in that position: it carries Appendix M, and the milder $\nu = 0.05$ instance the appendices also report is a different task that the harness names `burgers1d:0.05`.

**`allencahn1d` is excluded because it is saturated, and we ran the ladder that establishes it.** The ladder is registered in `notes/allencahn_ladder_design.md` at the same 64 seeds, budget, architecture and step count as every other ladder here, with its void rule and two competing predictions written down before it ran. The held uniform control fails $\mathbf{0/64}$ at *every* rung, at a median $R^2$ of 0.9997, and every arm, ours and the published ones, reaches 1.0000.

**The two-sided void rule fires along the whole ladder**, which is what it is for: a control that fails nothing cannot be crossed, so the failure axis reports nothing here and the paired mean against the re-drawn control becomes the statistic. The largest such mean over all thirty-five cells is 0.008, against a control whose own mean is 0.99953 at every rung because neither control reads $V$. **Both registered predictions are refuted by their letters and the substance is a null.** The mechanism's prediction asked for a within-rule slope significantly negative under Holm over the seven arms and none is, the smallest adjusted $p$ being 0.23. The suppression defence asked that no arm separate from uniform at any rung, and seven of the thirty-five cells are significantly below the control under Holm; the largest of them is $-0.008$, and fifteen cells are significantly *above* it by at most $+0.0005$. A task this well solved cannot discriminate between a mechanism and its suppression, which is why the rule is read before the arms rather than after.

That is the same ground this appendix gives for `convdiff1d`, and it is not the ground of an earlier version of this paragraph, which called the task optimisation-limited under Adam on the strength of a five-seed run whose uniform median was 0.530. Those five seeds now read 0.9999, 0.9997, 0.9999, 0.9998 and 0.9999.

**A five-seed number from an early run did not reproduce, and the reason is a limit on every early number in this paper.** We eliminated the candidates first. The task definition has not changed since that run. The harness has not either: that commit's own code, checked out and run, gives 0.99994 in test $R^2$ on the seed the run recorded at $-0.1535$. The diffusion constant is not the cause, at 0.01, 0.001 or 0.0001, and neither is the pool size, since that run's full-pool arm selected 1,200 points. What remains is the run's own record: its `metadata.json` carries seven fields and none of the ones that would settle it, with no pool size, no test-grid size and no PDE block. It predates the change that made each worker report the configuration it actually used, which was made after the old writer was found printing `lr=0.01` for runs that trained at 0.001. **So we cannot say what that run trained at, and a reader should discount any figure this paper quotes from a run of that vintage in the same way.** The runs behind every ladder, table and figure in the main text postdate it.

**These two exclusions were judged after the runs, and we say so for the same reason we say it about `staircase`.** The `allencahn1d` and `convdiff1d` runs are from June; the only design notes that mention either concern a later parameterised `convdiff1d` testbed built in August, and neither states a seed-stability or headroom criterion for the June runs. So the reasons above are ours, given afterwards, and a reader should weigh them as such. They are why the two-sided void rule of Section 2 now applies before any arm is read, and both tasks now fail it at the same end. `convdiff1d` fails the check because at 5 seeds *nothing* fails and uniform reaches 0.948, so there is no room for a rule to win or lose, which is the same condition that made `staircase` uninformative in Appendix Z and the same condition the 64-seed ladder finds on `allencahn1d`.

**Optimiser.** Adam only, 4000 steps, no L-BFGS. Neither task above sits out because of it any longer: both are out for saturation, which a stronger optimiser would deepen rather than remove. Adding L-BFGS would change which problems sit in the selection-limited regime and is an extension rather than a correction.

**The mechanism on `poisson1d`.** We can reproduce the effect reliably and cannot explain it. Two candidate mechanisms were tested; one was confounded and one was refuted.

**Adaptive re-selection cadence.** $c$ is fixed at 100 steps throughout. A schedule that re-selects more often early is an obvious variant and is not evaluated here.

**Larger networks and 2D-spatial domains.** See Section 7. Neither is a compute limit (a full 2D-spatial block is one overnight run on the machine used here) and both are simply not done.