PROJECTION ORDER MATTERS: LEARNING FROM UNREACHABLE TARGETS

Anonymous authors
Paper under double-blind review

ABSTRACT

Robot faces learning from human mouth motion, prostheses from able-bodied motion capture and impaired musculoskeletal models from a healthy expert learn from targets they cannot reach. Existing methods filter such demonstrations or retarget each onto the feasible set before training. When every target is unreachable, nothing survives filtering, and it is unclear whether to project targets before or after learning. This order matters through two competing effects; we make three contributions. (1) An exact finite-rank theorem: when the reachable set is a linear subspace, projecting first keeps unreachable directions from consuming a rank-limited linear student's capacity. (2) On curved reachable sets, projection and conditional averaging do not commute, so projecting noisy targets biases the learning target; constrained-Bayes distillation (CBD) averages with an offline teacher, projects the estimated mean and distils it into a small deployment model. In controlled sweeps, CBD improves on project-then-fit at every curvature, raw fitting is the best of the three orders when the student is wide and the set mildly curved, and no learned recipe wins everywhere. (3) The inverse that projects a target also returns commands, useful supervision even where the projected outputs are worse labels. On a simulated saturating trajectory plant, pretraining a feasible-output student on them turns the worst learner into the best at smaller widths, a gain compute-matched alternatives do not reproduce; its advantage over them replicates on a simulated two-link arm in a predeclared design. Projection order is thus a design choice, not a convention.

1 INTRODUCTION

A silicone robot face is trained to move like the human mouths in a video corpus. Its 40 tracked lip points span an 80-dimensional output space, its actuators reach a small bounded region of it, and most human frames lie outside that region (Figure 1). The face cannot produce its targets; neither can a prosthesis trained on able-bodied motion capture or an impaired musculoskeletal model imitating a healthy expert. Existing work recognizes this embodiment mismatch and filters demonstrations, estimates their feasibility, or retargets trajectories before training (Lee et al., 2020; Cao et al., 2022; Qiu et al., 2023). These remedies do not settle the case in which every target is unreachable: filtering then leaves nothing, and retargeting each demonstration is one particular order of operations, project then fit, whose statistical effect on a capacity-limited learner has not, to our knowledge, been analysed. To ask which target such a learner should be trained on when the reachable set is known only through a sampled or simulated plant, we compare the orders in which fitting, averaging and projection onto the reachable set can be taken, prove when the order matters, propose a distillation method that takes the order the theory favours, and show that the inverse used to project a target also supplies a useful training signal (contributions listed below).

Let X denote context, $Y \in \mathbb{R}^d$ a demonstrated target, and $\mathcal{R} \subseteq \mathbb{R}^d$ the output set realizable by the deployment system. Three natural orders of operation, and a fourth that fits inside a hypothesis class

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.

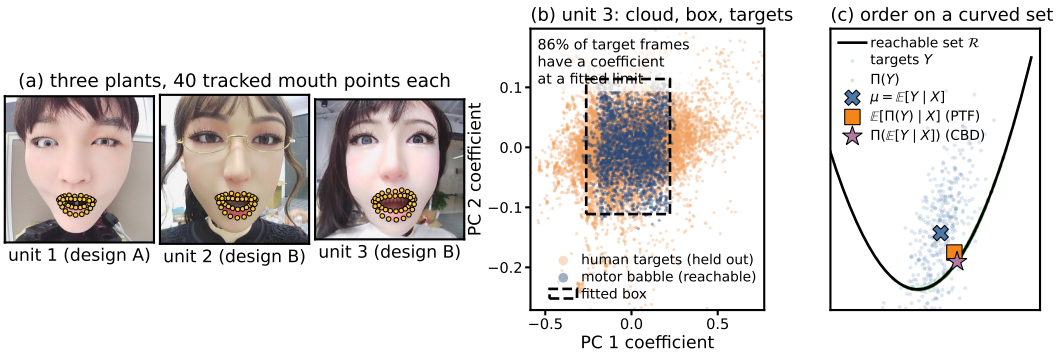


Figure 1: **Learning from targets the body cannot produce.** (a) Three silicone robot faces, two builds of one design and one of another, with the 40 tracked mouth points that define the 80-dimensional output space. (b) The motor-babble output cloud of one unit in its first two principal coordinates, the coefficient box fitted to it, and 200 held-out human target clips: on 86% of target frames at least one fitted limit is binding. (c) On a curved reachable set, the mean of the projected targets (PTF) and the projection of the mean (CBD) differ, and a final deployment projection does not remove the difference. Geometry only (Appendix C.6).

already respecting the constraint, are

$$\text{RAW} : Y \longrightarrow \text{fit} \longrightarrow \text{project}, \tag{1}$$

$$\text{PTF} : Y \longrightarrow \text{project} \longrightarrow \text{fit}, \tag{2}$$

$$\text{CBD} : Y \longrightarrow \text{estimate } \mathbb{E}[Y \mid X] \longrightarrow \text{project} \longrightarrow \text{distil}, \tag{3}$$

$$\text{EMBED} : \text{fit } f \in \mathcal{F}_C \text{ subject to } f(x) \in \mathcal{R} \text{ for every } x, \tag{4}$$

with $\mathcal{F}_C$ the student's hypothesis class of capacity C. We call a student trained in the last order, whose architecture ends in a model of the plant so that every output it emits is reachable, a *feasible-output student*; an analytic feasible parameterization (Gupta et al., 2026) is an exact instance. The differences are not cosmetic. When the reachable set is a linear subspace and the student has limited rank, fitting the full target may spend that rank on directions that deployment then discards; PTF prevents this *capacity drain*. When the reachable set is not a linear subspace, being curved or bounded by limits, projection and conditional expectation need not commute,

$$\mathbb{E}[\Pi_{\mathcal{R}}(Y) \mid X] \;\neq\; \Pi_{\mathcal{R}}(\mathbb{E}[Y \mid X]) \quad \text{in general}, \tag{5}$$

so projecting each noisy observation biases the population target (Figure 1c). CBD averages first: a large offline teacher estimates the conditional mean, the estimate is projected through an offline plant model, and a small deployment model is distilled on these feasible labels; unlike knowledge distillation (Hinton et al., 2015), what the teacher changes is the order of averaging and projection, not the capacity of the student.

Our contributions are as follows.

1. **Projecting first can protect capacity.** When the reachable set is a linear subspace and a linear student's rank is limited, we prove that training on projected targets (PTF) is never worse than raw fitting in the error that survives deployment, state exactly when it is strictly better, and show that giving the student more rank than the reachable dimension does not by itself remove the loss (Section 3).
2. **On a curved or bounded set, averaging first changes the target.** We derive the best feasible predictor, show that PTF is biased away from it because averaging and projection do not commute, propose CBD, which averages first with an offline teacher, and bound its error by student, projector and teacher terms (Section 4). On a controlled curved benchmark CBD improves on PTF at every curvature, raw fitting is the best of the three orders when the student is wide and the curvature low, and a feasible-output student with an exact plant model has the best pooled mean of the methods that do not see the true conditional mean (Section 5.1).

3. **One inverse solve, two kinds of supervision.** Projecting a target through a plant model returns a feasible output and the commands that realize it. On a simulated actuator with saturation and a dead zone (Section 5.2) the projected outputs are the worse regression targets, yet pretraining a feasible-output student on the commands turns it from the worst method under a random start into the best at widths 8 and 16. Extra updates, a tuned multi-start or curriculum on the same plant budget, and imitating the realized outputs at twice the training-stage plant cost do not reproduce the gain; the advantage over the multi-start and the curriculum replicates on a simulated two-link arm in a design fixed before the run.

All learning experiments run in simulation. The three physical faces of Figure 1 are used only to measure the geometry of real reachable sets (Appendix C.6), and an independent re-implementation checks the curved benchmark (Appendix E).

2 Problem formulation

For each context x, a predictor must ultimately produce an output in the closed reachable set $\mathcal{R}$. Unlike feasibility-checking settings, in which a demonstration either admits a constraint-compliant execution or is rejected (Gupta et al., 2026), we assume $\Pr[Y \notin \mathcal{R}] > 0$, and typically $Y \notin \mathcal{R}$ almost surely, so that every predictor must approximate. We evaluate squared realized-output loss

$$\mathcal{L}(h) = \mathbb{E}\left[\|Y - h(X)\|_2^2\right], \qquad h(X) \in \mathcal{R}. \tag{6}$$

Let $\mu(x) = \mathbb{E}[Y \mid X = x]$, and let $\Pi_{\mathcal{R}}(y)$ be a selected Euclidean nearest point in $\mathcal{R}$.

Proposition 1 (Constrained Bayes rule)**.** *For every x for which $\Pi_{\mathcal{R}}(\mu(x))$ is unique, $h^\star(x) = \Pi_{\mathcal{R}}(\mu(x))$ minimizes $\mathbb{E}[\|Y - h\|_2^2 \mid X = x]$ over $h \in \mathcal{R}$.*

The proof is the conditional bias–variance decomposition (Appendix A). A physical implementation may realize the projection using commands,

$$u^\star(y) \in \arg\min_{u \in \mathcal{U}} \|y - \widehat{\Phi}(u)\|_W^2 + \Omega(u), \qquad \widehat{\Pi}(y) = \widehat{\Phi}(u^\star(y)), \tag{7}$$

where $\widehat{\Phi}$ is an offline plant model or simulator, Ω contains command regularizers, and the optimizer may be gradient based or black box. With an exact plant model and a global minimizer, $W \propto I$ and $\Omega \equiv 0$ return the Euclidean nearest reachable output; a weighted norm, a command regularizer or a local optimum can return something else. Throughout, $\Pi_{\mathcal{R}}$ is the exact projector the analysis is written against and $\widehat{\Pi}$ the operator actually applied; ϵ_P in Proposition 3 bounds their difference.

3 Capacity drain in linear reachable subspaces

Translate coordinates so that an affine reachable set passes through the origin; it is then a linear subspace, onto which $P = P^\top = P^2$ projects, and $Q = I - P$. A free intercept absorbs the constant part of the conditional mean under both orders (Appendix A.2), so assume $X \sim \mathcal{N}(0, I)$, $\mu(X) = MX$ and a linear student with $\operatorname{rank}(W) \le r$. Population raw regression returns a best rank-r approximation M_r of M; PTF returns a best rank-r approximation $(PM)_r$ of PM.

Theorem 2 (Finite-rank capacity drain)**.** *Let reachable deployment error be $\mathcal{L}_R(W) = \mathbb{E}\|PMX - PWX\|_2^2$. Then*

$$\mathcal{L}_R((PM)_r) = \|PM - (PM)_r\|_F^2 \le \|PM - PM_r\|_F^2 = \mathcal{L}_R(M_r). \tag{8}$$

The inequality is strict exactly when PM_r is not a best rank-r approximation of PM.

The theorem follows directly from the Eckart–Young–Mirsky result (Eckart & Young, 1936; Mirsky, 1960); what it adds is a reading for deployment. Raw regression ranks the singular modes of $M = PM + QM$ jointly, so a large mode of QM can enter the top-r approximation, then vanish under deployment projection, while an omitted mode of PM creates irrecoverable realized error. PTF ranks only modes that survive deployment (Appendix A.2, Figure 5). PTF is exact once $r \ge \operatorname{rank}(PM)$; RAW is exact only when $PM_r = PM$, for which $r \ge \operatorname{rank}(M)$ suffices, and surplus rank over the reachable dimension does not by itself remove the drain: with $M = \operatorname{diag}(1, 3, 2)$,

$P = \mathrm{diag}(1, 0, 0)$ and $r = 2 > \mathrm{rank}(PM) = 1$, raw regression keeps the two unreachable modes and pays reachable error 1 where PTF pays none (Appendix A.2). The theorem assumes linear geometry, squared loss, population optimization and a rank-limited linear student; a low-dimensional output cloud such as Figure 1b describes the reachable set, not the student's rank r, so it does not show that the drain is absent.

4 Nonlinear geometry and constrained-Bayes distillation

For a nonlinear $\mathcal{R}$, PTF trains toward $m_{\mathrm{PTF}}(x) = \mathbb{E}[\Pi_{\mathcal{R}}(Y) \mid X = x]$, the function it would learn with unlimited capacity and data. If $m_{\mathrm{PTF}}(x) \notin \mathcal{R}$, deployment applies $\Pi_{\mathcal{R}}$ once more, and neither m_{PTF} nor $\Pi_{\mathcal{R}}(m_{\mathrm{PTF}})$ generally equals $h^\star = \Pi_{\mathcal{R}}(\mu)$. Locally, when the selected projector is twice differentiable at $\mu(x)$, as it is within the reach of a smooth $\mathcal{R}$, the discrepancy is second order in the noise scale: for a shrinking family $Y_\tau = \mu + \tau\varepsilon$ with $\mathbb{E}[\varepsilon \mid X = x] = 0$ and $\Sigma(x) = \mathbb{E}[\varepsilon\varepsilon^\top \mid X = x]$,

$$\mathbb{E}[\Pi_{\mathcal{R}}(Y_\tau) \mid X = x] = \Pi_{\mathcal{R}}(\mu(x)) + \tfrac{\tau^2}{2} H_\Pi(\mu(x)) : \Sigma(x) + o(\tau^2), \tag{9}$$

where $(H_\Pi : \Sigma)_k = \mathrm{tr}(\nabla^2 \Pi_k \Sigma)$ (Appendix A). The leading discrepancy couples reachable-set curvature and conditional uncertainty. A deployment projection removes its component normal to $\mathcal{R}$ to leading order, so a deployed output student inherits the tangential part; Figure 1c shows it on a parabola, and Table 10 computes both parts by Gauss–Hermite quadrature on the benchmark of Section 5.1.

Non-smooth boundaries are the common case. Actuators have range limits. If each output coordinate has its own limits, the reachable set is a box and projecting onto it clips each coordinate. Eq. 9 fails at a limit, where clipping is not differentiable, yet clipping and averaging still do not commute: where the noise straddles a limit the discrepancy is first order in the noise scale σ rather than second (Figure 2; Appendix A.3). With Gaussian noise and the conditional mean exactly on an upper limit, clipping every sample and then averaging lands $\sigma/\sqrt{2\pi}$ inside the limit, while averaging and then clipping gives the optimum, the limit itself. The discrepancy is largest when the mean sits on a limit, decays within a few noise scales on either side, and vanishes when every sample falls on the same side of each limit: a noise-free target beyond a limit is clipped on every sample and has no bias, so what matters is the mean's distance to the limit on the scale of the noise, not the fraction of samples clipped. A PTF student, trained on clipped samples, therefore learns outputs pulled inside the limits exactly where the demonstrations press against them, at the extremes of the behaviour. Limits on the *commands* rather than on the outputs do not make the output set a box; the trajectory plant of Section 5.2 is an example.

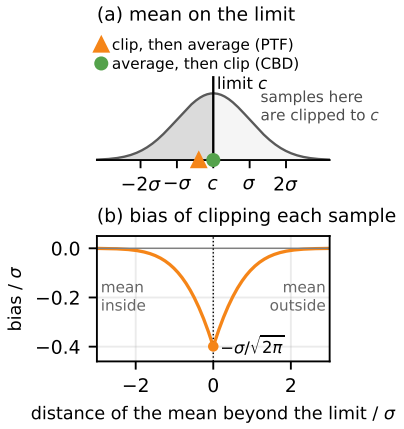


Figure 2: **Clipping and averaging do not commute** (one upper limit c, Gaussian noise of scale σ). (a) Mean on the limit. (b) Bias of clipping each sample against the mean's distance beyond the limit (Eq. 18).

CBD algorithm. The three orders train the same student with the same loss and differ only in the label each example receives: RAW uses y_i, PTF uses $\widehat{\Pi}(y_i)$, and CBD uses $\widehat{\Pi}(T_\psi(x_i))$, the projection of a teacher's estimate of the conditional mean at x_i. With exact projection and over the noise at a given x, PTF's labels average to $\mathbb{E}[\Pi_{\mathcal{R}}(Y) \mid x]$, which Eq. 5 separates from the optimum $h^\star(x) = \Pi_{\mathcal{R}}(\mu(x))$; CBD's labels approach $h^\star(x)$ as the teacher improves, at the price of an offline teacher and one projection per training example. Concretely, fit an offline teacher $T_\psi(x) \approx \mu(x)$, either on all training pairs or cross-fitted so that no label is predicted by a model that saw it (the sweeps run both); construct cached targets $z_i = \widehat{\Pi}(T_\psi(x_i))$ using Eq. 7; and fit the deployment student $\widehat{f} \in \arg\min_{f \in \mathcal{F}_C} \sum_i \|f(x_i) - z_i\|^2$. Teacher fitting and target construction are offline; at deployment the student runs, together with the realization map of Table 3 (Appendix B.1) wherever it emits outputs rather than commands, exactly as for RAW and PTF. With an oracle teacher, $T_\psi = \mu$ (available only in simulation; we run it as the *oracle CBD* arm), and an exact projector, $z(x) = h^\star(x)$,

so CBD removes both unreachable target variation and the population projection-order bias; a learned implementation inherits new error sources.

Proposition 3 (Local CBD error decomposition)**.** *Let g denote the mapping actually deployed, the feasible output that the deployment realization map produces from the student (Table 3); exact deployment is the case $g = \Pi_{\mathcal{R}}(\widehat{f})$. Suppose that on a neighbourhood of $\mu(x)$ and $T_\psi(x)$ the projector $\Pi_{\mathcal{R}}$ is L_Π-Lipschitz and $\|\widehat{\Pi}(v) - \Pi_{\mathcal{R}}(v)\| \le \epsilon_P$ for every v in it, and that $\|T_\psi(x) - \mu(x)\| \le \epsilon_T(x)$. If the deployed output satisfies $\|g(x) - \widehat{\Pi}(T_\psi(x))\| \le \epsilon_S(x)$, then*

$$\|g(x) - h^\star(x)\| \le \epsilon_S(x) + \epsilon_P + L_\Pi \epsilon_T(x). \tag{10}$$

The bound says what CBD needs: its deployed output approaches the optimum $h^\star$ to the extent that three distinct errors are small, the student's fit to its cached labels, the offline projector's error, and the teacher's error, amplified by L_Π where the projection is ill-conditioned. Section 5.1 probes each in turn (student width, teacher data against teacher width, perturbed projectors). It also says why an oracle target need not make learned CBD the best method: teacher error, labels that are hard for a small student to represent, or a projector that is wrong at deployment can outweigh the bias that CBD removes. It is an upper bound (Appendix A.4 gives the reach condition behind L_Π and the pre-projection form of ϵ_S), and replacing the target also changes the student's approximation and optimization problem, so a measured difference between two arms is at most *consistent with* one term. The limiting alternative is EMBED, which needs no deployment inverse but a differentiable plant model during training (Table 3); Section 5.2 shows on two plants when it is the best choice and what it takes to make it one.

5 Experiments

The experiments test the three contributions where each can be isolated. Section 5.1 uses a curved reachable set with an exact projector, where the projection-order bias can be computed (contribution 2); Section 5.2 a simulated actuator with saturation and a dead zone, where projecting means running an iterative inverse through the plant, and a simulated two-link arm (contribution 3); Section 5.3 a muscle-driven MyoSuite model, the geometry of the three physical faces and an independent re-implementation. Theorem 2 is exact and illustrated in Appendix A.2. All learning runs are CPU simulations. Each writes one row per (arm, cell, seed), an *arm* being one training method and a *cell* one setting of the swept variables, and a released script makes every table and figure from those rows (Appendix F). The static and trajectory sweeps use fifteen seeds, the other seeded experiments five unless stated otherwise. A comparison between two arms is a paired difference over seeds, pooled over cells by averaging within each seed first, and is *decided* when it is statistically significant: the paired mean exceeds $t_{0.975,n-1}$ standard errors (2.14 for fifteen seeds), a two-sided 95% paired t-test. Four tests were *predeclared*: chosen from a five-seed pilot and fixed before the fifteen-seed run, so that they confirm rather than select (Appendix C.1). Summaries are mean ± standard deviation, paired contrasts mean difference ± standard error over seeds.

Two evaluation criteria. Each row records two numbers for the deployed map g. The *excess task risk* $\mathcal{L}(g) - \mathcal{L}(h^\star) = \mathbb{E}\|g - \mu\|^2 - \mathbb{E}\|h^\star - \mu\|^2$ is what the system is judged by; the *deployed Bayes-target distance* $D_B = \mathbb{E}\|g - h^\star\|^2$ measures how closely a method recovers the constrained Bayes predictor that the theory targets. By an exact identity (Appendix C.1) they differ by $2\mathbb{E}\langle g - h^\star, h^\star - \mu\rangle$, which vanishes on affine sets but not on curved or bounded ones, so the two can rank methods differently. Measured, they rank RAW, PTF and CBD identically in all 36 cells of the curved benchmark (Table 7) and disagree on some comparisons on the trajectory plant (Tables 26 and 30). Prose and figures give excess task risk unless labelled D_B; the appendix repeats every display under D_B. All errors are means per output coordinate, over test inputs and over the 2 coordinates of the curve or the $24 \times 2 = 48$ numbers of a trajectory.

5.1 Controlled nonlinear benchmark

The reachable set is a parabola $r(u) = (u, \frac{c}{2}u^2)$ with an exact nearest-point projector, the conditional mean is displaced from it along the curve's normal by an input-dependent amount, and the conditional noise is anisotropic, correlated and input dependent, so that projection and expectation do not

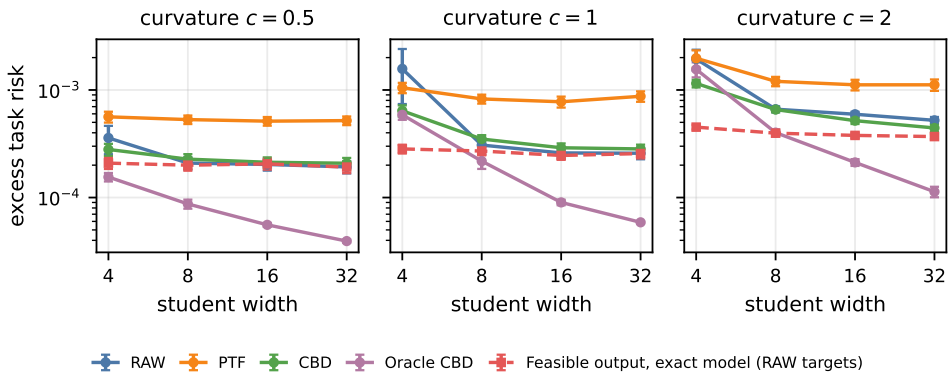


Figure 3: **Capacity and curvature at high conditional noise** ($\sigma = 0.4$): fifteen-seed means of excess task risk with standard-error bars. Dashed: the feasible-output student with the exact model (EMBED), the best pooled non-oracle arm. Best-mean maps over all 36 cells and the D_B displays are in Appendix C.2.

commute by accident. We vary student width $\{4, 8, 16, 32\}$, curvature $c \in \{0.5, 1, 2\}$ and conditional noise $\sigma \in \{0.05, 0.2, 0.4\}$, with 2,048 training contexts, fifteen seeds and 16,200 seed-level rows, and compare RAW, PTF, learned CBD with an in-sample and with a cross-fitted teacher, oracle CBD, partial PTF, standard distillation, a soft constraint, and architecture-matched feasible-output students (EMBED) whose differentiable model carries a relative curvature error in $\{-0.5, -0.25, 0, 0.25, 0.5\}$.

On this grid, CBD improves on PTF at every curvature, and RAW has the best mean of the three orders in every cell with a student of width 8 or more at the two lower curvatures (18 of 18; Figure 3). Curvature thus determines which order has the best mean in most cells, and capacity whether that lead is statistically decided: the best-mean order is the same under both criteria in every cell, and 19 task-risk and 18 Bayes-distance cells are decided (Table 7). Every decided cell goes to the order the mechanism predicts. RAW takes thirteen, all at the two lower curvatures and widths 8 to 32, where the student can recover the conditional mean and the deployment projection is benign; CBD takes six, each at the highest curvature, where the discrepancy of Table 10 is largest, or at the narrowest width, where the student is least able to recover the conditional mean; PTF's two best-mean cells are not decided. Pooled over the grid with the seed as the unit, CBD beats PTF by $2.1 \times 10^{-4} \pm 0.4 \times 10^{-4}$ ($|t| = 4.7$), and also when pooled within each curvature (Table 8), consistent with the population discrepancy of Table 10 being removed from the target, whereas neither PTF nor CBD is decided against RAW pooled. The two predeclared CBD-versus-PTF contrasts, at $c = 2$, $\sigma = 0.4$ and widths 16 and 32, are decided in CBD's favour and the predeclared RAW-favourable control (RAW against CBD at $c = 1$, $\sigma = 0.05$, width 32) is decided for RAW (Table 5); all four predeclared tests also remain decided on the ten seeds run after the choice.

Three further readings separate target quality, estimation data and model error. Oracle CBD, which uses the true conditional mean, beats learned CBD in 365 of 540 cell-seed pairs but loses to it at width 4: the target that is best on average over the data need not be the one a small student learns best. The feasible-output student (EMBED, Eq. 4) with the true curvature inside it has the best pooled mean of the non-oracle arms ($1.3 \times 10^{-4} \pm 0.1 \times 10^{-4}$ against learned CBD, the predeclared pooled contrast), but a 25% error in its embedded curvature already puts it 20 to 45 times above offline CBD, which never trains through that model (Appendix C.2). More teacher data helps where a larger teacher does not: an eight-fold reduction of teacher error across 256 to 4,096 contexts buys a six-fold reduction of CBD's excess risk, while teacher width buys none (Appendix C.3). Perturbing the offline projector and the embedded model gives mirror-image sensitivities, and a robust projector ensemble is worse than nominal CBD in all 125 paired comparisons (Appendix C.4).

5.2 Non-smooth trajectory plant

To test the orders where no closed-form projector exists, we use a simulated actuator (Appendix C.5). A context x gives the amplitude, frequency and phase of a desired two-channel oscillation over 24 time

Table 1: **Trajectory plant**, fifteen seeds, mean ± std. of excess task risk and D_B against the reference $\tilde{h}$ ($\times 10^{-2}$), and the per-seed wall-clock cost at width 32 (itemized in Appendix C.5). Bold is the best mean per column. Feasible-output rows embed the exact plant, cold from a random start or warm from the hybrid's command imitation; all arms, the reference-free $\mathbb{E}\|g-\mu\|^2$ and the paired differences are in the same appendix.

	width 8		width 16		width 32		cost
Method ($\times 10^{-2}$)	task risk	D_B	task risk	D_B	task risk	D_B	(s)
RAW	4.41 ± 0.89	3.46 ± 0.66	1.42 ± 0.29	1.24 ± 0.21	0.64 ± 0.14	0.62 ± 0.11	1.8
PTF	5.84 ± 1.44	3.82 ± 0.94	2.51 ± 0.37	1.62 ± 0.24	1.33 ± 0.33	0.85 ± 0.23	3.2
CBD	6.68 ± 1.96	4.47 ± 1.38	2.11 ± 0.39	1.37 ± 0.26	1.19 ± 0.24	0.78 ± 0.16	8.6
Oracle CBD	5.41 ± 1.07	3.57 ± 0.72	2.22 ± 0.36	1.43 ± 0.23	1.21 ± 0.14	0.77 ± 0.11	3.2
Feasible, cold (RAW)	7.38 ± 1.49	5.33 ± 1.11	3.65 ± 0.79	2.74 ± 0.60	3.86 ± 1.37	2.90 ± 1.00	7.2
Feasible, cold (CBD)	6.22 ± 1.35	4.15 ± 0.91	3.33 ± 0.75	2.24 ± 0.52	2.71 ± 0.64	1.85 ± 0.47	14.1
Feasible, warm (RAW)	$\mathbf{1.73 \pm 0.27}$	1.34 ± 0.22	$\mathbf{0.75 \pm 0.06}$	0.54 ± 0.04	$\mathbf{0.57 \pm 0.04}$	0.39 ± 0.03	9.2
Feasible, warm (CBD)	1.85 ± 0.38	$\mathbf{1.28 \pm 0.30}$	0.83 ± 0.06	$\mathbf{0.52 \pm 0.05}$	0.63 ± 0.04	$\mathbf{0.38 \pm 0.03}$	14.5

steps, with the amplitude scaled up by 45%, which puts the conditional mean outside the reachable set, and observation noise is added. The plant maps a sequence of two-channel commands in $[-1, 1]$ to an output trajectory through a command dead zone, a coupling limit between the channels, asymmetric rise and fall rates and a state carried between time steps; its rate switch is a continuous ramp, so its reachable set is compact, as the nearest-point projection of the theory requires (Proposition 5). Students of width 8, 16 and 32 are trained on 512 contexts, with fifteen seeds. Projecting means a 120-step gradient search over commands through the plant; labels and deployment use the same inverse and budget (a smaller deployment budget would penalize RAW, whose requests lie furthest from the set). Both criteria are measured against a separately built numerical reference $\tilde{h}$, not the unknown Bayes optimum: paired task-risk differences do not depend on it, paired D_B differences and percentages of a gap do, and the reference-free $\mathbb{E}\|g-\mu\|^2$ orders the arms identically (Appendix B.2).

Among the standard-budget cold arms of Table 1, RAW has the best mean at every width under excess task risk and is decided against the runner-up at each, consistent with the mechanism at these widths: RAW recovers the conditional mean and the deployment inverse realizes it, whereas PTF retains projection-order bias and learned CBD teacher error. Oracle CBD has neither and still trails RAW. Measured against a diagnostic anchor (μ for RAW, $\widehat{\Pi}(\mu)$ otherwise), the deployment inverse reduces RAW's mean squared error to about half and the projected orders' only to 80 to 93%, and it displaces RAW's predictions seven to thirteen times as far in mean square (Appendix C.5). CBD beats PTF at width 16 (by 0.0040 ± 0.0014, decided) and is not distinguished from it at widths 8 and 32: CBD's decided wins in Section 5.1 all need high curvature or a narrower student than any run here. The two criteria do not always agree: three of the five contrasts in Figure 4b are decided under one and not the other.

Inverse-command pretraining unlocks the feasible student. The feasible-output students embed the true plant with no parameter error, yet a random initialization leaves them behind RAW by a factor of 1.4 to 1.7 at width 8 and 2.3 to 6.0 at widths 16 and 32, decided for every matched pair: an exact model of the plant is not enough on its own.

The commands that the projected orders compute while building their labels change this. The hybrid recipe is three steps on one command network q_θ: (i) run the inverse the projected orders already run and cache its commands u_i; (ii) fit $q_\theta(x_i)$ to u_i, with no plant in the loop; (iii) continue end to end through the embedded plant on the arm's own targets, with the usual budget. At deployment the network's commands drive the plant; the recipe trains the EMBED order, not a fifth statistical target. The warm-started feasible-output family then has the best mean at every width under both criteria (the RAW-target hybrid under task risk, the CBD-target one under D_B). The RAW-target hybrid is ahead of RAW by 0.0268 ± 0.0024 and 0.0067 ± 0.0007 at widths 8 and 16 (every seed); at width 32 its lead of 0.0008 ± 0.0004 under task risk is not decided, and under D_B it is ahead by 0.0023 ± 0.0003 in every seed. Figure 4a shows what the gain is made of. The frozen command-imitation network alone already beats the cold student in every seed; the end-to-end stage then cuts its error by a further factor of 1.6 to 2.3, and that stage is what puts the student ahead of RAW; a cold start with the same total number of updates stays 0.017 to 0.025 behind the hybrid at every width. So the result points to

optimization through the plant as the bottleneck, and inverse-command supervision gives the search a useful start from either label source (the two warm students' means differ by at most 0.0012).

An inverse solve returns both a feasible output and the commands that realize it: which should supervise the first stage? Fitting the same student, through the plant, to the outputs — same solve, same epochs, same step (iii), twice the training-stage plant calls — is worse than fitting the commands at every width, on both target families and in every seed (0.0149 ± 0.0018 pooled on raw targets, predeclared; Figure 4a), and already worse as a frozen first stage. It does beat a cold start, so what the commands buy is not having a first stage but a target the network can be fitted to with no plant in its loop at all (Table 29).

Is the gain just extra offline computation? Two alternatives that spend the same budget without imitating inverse commands do not reproduce it (Table 20; predeclared): a multi-start with learning-rate tuning and a curriculum from an easy plant without dead zone, coupling limit or rate asymmetry to the true plant, each given the hybrid's 721 plant-batch calls, exactly on the raw targets of the primary contrasts and more on CBD targets (Appendix C.5). Both improve on the cold student and both stay 0.026 ± 0.002 behind the hybrid, pooled over widths within seed, in every seed, Holm-corrected; the secondary CBD-target contrasts agree. The hybrid spends part of the plant budget on a per-example inverse search whose commands bypass the gradients through the dead zone; the alternatives spend all of it on those gradients. With a misspecified plant inside the feasible layer the hybrid still beats the cold student at every width and in every cell of the plant-error grid (0.044 ± 0.002 pooled, every seed, predeclared), and stays ahead of RAW at widths 8 and 16 but not at 32 (Table 28).

The trade is offline computation for a deployment that needs no inverse: charged the inverse it imitates, the warm student costs 15 to 18 times RAW offline, but deploys with one plant call per query batch, in milliseconds, where every output arm pays 122 calls and about a second (Appendix C.5).

The warm start replicates on a plant of a different kind: a simulated torque-driven two-link arm, observed at its end effector and driven in open loop by the student's torques, with rigid-body dynamics, damping, a torque limit and inelastic joint stops (which make its realization map discontinuous, outside Proposition 5). Its demonstrations come from a stronger arm of the same kinematics and so ask for more torque, less damping or more elbow range than it has; two regimes of it, fixed on plant-level criteria before any student was trained, leave the label projector 0.7 and 7.9 times the observation-noise variance from the conditional mean (*easy* and *hard*). With the design, its primary contrasts and sample-size rule fixed before a separate pilot, and thirty confirmatory seeds at two widths (Tables 22 and 23), the warm student beats the compute-matched multi-start by 0.0041 ± 0.0002 and the curriculum by 0.0111 ± 0.0008, pooled over the four conditions within seed, in every seed and every condition, Holm-corrected, and the secondary CBD-target contrasts agree. It beats the cold student in the easy regime as in the hard one, so demonstrations reachable to within the noise do not make cold training sufficient here. Below the warm start the arm differs from the first plant: the cold feasible student already beats RAW in three of the four conditions, the multi-start falls just short of the $1{,}200$-epoch cold student, and the curriculum is worse than cold training (Appendix C.5, with further robustness checks). What carries across the two plants is thus the warm start, not the ranking beneath it.

5.3 Three further tests, reported in the appendix

On MyoSuite's MyoElbow (Appendix C.7), with muscle commands for an impaired plant, the PTF and learned-CBD students track about three times better than healthy-command behaviour cloning, a test of command-space targets rather than of the output-space theorem. The three faces' fitted box (Appendix C.6; supplementary video) is an estimate, not a certified reachable set, and the recordings cannot measure the bias itself. An independent re-implementation with isotropic noise (Appendix E) favours RAW in 31 to 36 of 36 cells: its larger projection-order gap is mostly normal to the curve, so less of it survives deployment (in 7 of 9 matched cells) than the teacher costs. A benchmark's noise structure is part of its result.

6 Related work

Feasibility-aware and out-of-dynamics imitation filter, weight or model demonstrations from agents with different dynamics (Lee et al., 2020; Cao et al., 2022; Qiu et al., 2023), and retargeting or

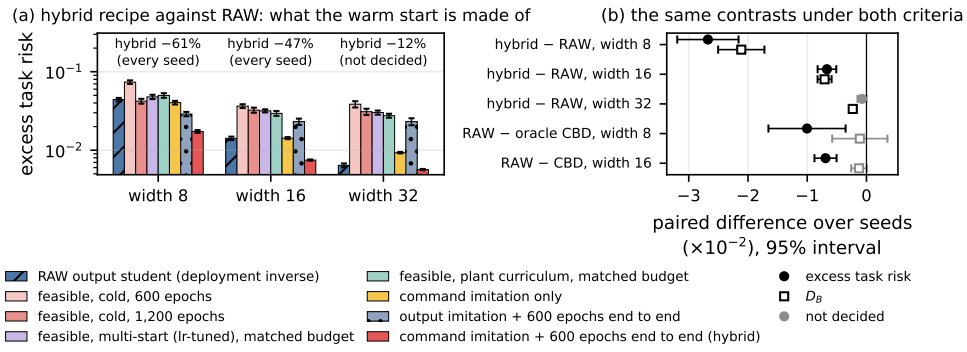


Figure 4: **What the warm start is made of** (trajectory plant, fifteen seeds). (a) Excess task risk of the RAW output student and of the feasible-output student on the same targets under seven recipes: cold for 600 epochs, cold for 1,200 (the hybrid's updates), a multi-start and a plant curriculum on the hybrid's 721 plant-batch calls, the frozen command-imitation network alone, *output* imitation plus the same end-to-end stage, and the hybrid; seed means and standard errors. (b) Paired differences under excess task risk (filled) and D_B (hollow), 95% intervals; grey: not decided. Other arms: Appendix C.5.

action-constrained imitation (Yeh et al., 2025) builds a feasible surrogate data set before policy learning; in our terms these are the PTF order applied per demonstration or a reweighted RAW target, and we compare the orders, not those pipelines. Optimizer-generated supervision is established (Levine & Koltun, 2013; Liu et al., 2022); we reuse commands the projected orders already compute. Architectures with built-in output constraints (Amos & Kolter, 2017; Agrawal et al., 2019; Brosowsky et al., 2021) and certified feasible controllers (Gupta et al., 2026) are the EMBED order and our strongest control. Extrinsic and Fréchet regression (Lin et al., 2017; Petersen & Müller, 2019) take conditional means onto or on a manifold, but their responses already lie on it; reduced-rank regression (Anderson, 1951; Izenman, 1975) is the classical account of which Theorem 2 is the deployment-projected reading; decision-focused learning (Donti et al., 2017; Elmachtoub & Grigas, 2022) changes the criterion, where we change the target; and CBD differs from knowledge distillation (Hinton et al., 2015; Hu et al., 2016) in its defining object, the constrained Bayes target (full comparison in Appendix H).

7 Discussion and Conclusion

No learned recipe won throughout: raw fitting was strongest wherever capacity and data sufficed, learned CBD where curvature and noise were both large, PTF nowhere decided, and the two trajectory plants disagreed about the cold feasible student. The rule they suggest — exact feasible parameterization when it is accurate and started in the right basin, CBD when the projection-order bias is material and teacher and projector are controlled, RAW when capacity and data suffice and an inverse can run at deployment — explains what we observed; we claim no validated selection rule. Three limitations bound the claims: the regime maps cannot rule out an intermediate algorithm (a half dose of projection already beats both endpoints in grand mean; Appendix D); under a perceptual criterion the optimal feasible target is no longer $\Pi_{\mathcal{R}}(\mathbb{E}[Y \mid X])$; and the geometry that makes the bias possible is measured on physical plants while the conditional quantities that make it large are not, so a learned system on a box-bounded plant with repeated-condition data is the missing measurement.

Unreachable supervision creates two distinct effects, finite-capacity competition and nonlinear projection-order bias. PTF removes the first in linear subspaces; CBD targets the constrained Bayes predictor, the population optimum for squared realized-output loss but not always what a finite student is best trained to produce. On the simulated trajectory plant, one inverse solve gave two training signals: its feasible outputs define the targets, and its commands are the better start for end-to-end learning.

REFERENCES

Akshay Agrawal, Brandon Amos, Shane Barratt, Stephen Boyd, Steven Diamond, and J. Zico Kolter. Differentiable convex optimization layers. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Brandon Amos and J. Zico Kolter. OptNet: Differentiable optimization as a layer in neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 136–145. PMLR, 2017.

T. W. Anderson. Estimating linear restrictions on regression coefficients for multivariate normal distributions. *The Annals of Mathematical Statistics*, 22(3):327–351, 1951. doi: 10.1214/aoms/1177729580.

Mathis Brosowsky, Florian Keck, Olaf Dünkel, and Marius Zöllner. Sample-specific output constraints for neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 6812–6821, 2021. doi: 10.1609/aaai.v35i8.16841.

Vittorio Caggiano, Huawei Wang, Guillaume Durandau, Massimo Sartori, and Vikash Kumar. MyoSuite: A contact-rich simulation suite for musculoskeletal motor control. *arXiv preprint arXiv:2205.13600*, 2022.

Zhangjie Cao, Yilun Hao, Mengxi Li, and Dorsa Sadigh. Learning feasibility to imitate demonstrators with different dynamics. In *Proceedings of the 5th Conference on Robot Learning*, volume 164 of *Proceedings of Machine Learning Research*, pp. 363–372. PMLR, 2022.

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

Priya L. Donti, Brandon Amos, and J. Zico Kolter. Task-based end-to-end model learning in stochastic optimization. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Carl Eckart and Gale Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, 1936.

Adam N. Elmachtoub and Paul Grigas. Smart "Predict, then Optimize". *Management Science*, 68(1): 9–26, 2022. doi: 10.1287/mnsc.2020.3922.

Herbert Federer. Curvature measures. *Transactions of the American Mathematical Society*, 93(3): 418–491, 1959.

Sthithpragya Gupta, Durgesh Haribhau Salunkhe, and Aude Billard. Demonstrate once, execute on many: Kinematic intelligence for cross-robot skill transfer. *Science Robotics*, 11(113):eaea1995, 2026. doi: 10.1126/scirobotics.aea1995.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Zhiting Hu, Xuezhe Ma, Zhengzhong Liu, Eduard Hovy, and Eric Xing. Harnessing deep neural networks with logic rules. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2410–2420. Association for Computational Linguistics, 2016.

Alan Julian Izenman. Reduced-rank regression for the multivariate linear model. *Journal of Multivariate Analysis*, 5(2):248–264, 1975. doi: 10.1016/0047-259X(75)90042-1.

Youngwoon Lee, Edward S. Hu, Zhengyu Yang, and Joseph J. Lim. To follow or not to follow: Selective imitation learning from observations. In *Proceedings of the Conference on Robot Learning*, volume 100 of *Proceedings of Machine Learning Research*, pp. 11–23. PMLR, 2020.

Sergey Levine and Vladlen Koltun. Guided policy search. In *Proceedings of the 30th International Conference on Machine Learning (ICML 2013)*, volume 28 of *Proceedings of Machine Learning Research*, pp. 1–9, 2013.

Lizhen Lin, Brian St. Thomas, Hongtu Zhu, and David B. Dunson. Extrinsic local regression on manifold-valued data. *Journal of the American Statistical Association*, 112(519):1261–1273, 2017. doi: 10.1080/01621459.2016.1208615.

I-Chun Arthur Liu, Shagun Uppal, Gaurav S. Sukhatme, Joseph J. Lim, Peter Englert, and Youngwoon Lee. Distilling motion planner augmented policies into visual control policies for robot manipulation. In *Proceedings of the 5th Conference on Robot Learning (CoRL 2021)*, volume 164 of *Proceedings of Machine Learning Research*, pp. 641–650, 2022.

Leon Mirsky. Symmetric gauge functions and unitarily invariant norms. *The Quarterly Journal of Mathematics*, 11(1):50–59, 1960.

Alexander Petersen and Hans-Georg Müller. Fréchet regression for random objects with Euclidean predictors. *The Annals of Statistics*, 47(2):691–719, 2019. doi: 10.1214/17-AOS1624.

Yiwen Qiu, Jialong Wu, Zhangjie Cao, and Mingsheng Long. Out-of-dynamics imitation learning from multimodal demonstrations. In *Proceedings of the 6th Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pp. 1071–1080. PMLR, 2023.

Chia-Han Yeh, Tse-Sheng Nan, Risto Vuorio, Wei Hung, Hung-Yen Wu, Shao-Hua Sun, and Ping-Chun Hsieh. Action-constrained imitation learning. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, volume 267 of *Proceedings of Machine Learning Research*, 2025. arXiv:2508.14379.

USE OF LARGE LANGUAGE MODELS

An LLM-based assistant was used throughout the preparation of this manuscript and of its software. It drafted and edited prose, proposed related-work positioning and references, and helped to formulate, scope and correct analytical claims, among them several of the statements, restatements and counterexamples of Sections 3 and 4 and of their appendices. It wrote and revised the released benchmark code, the drivers that run the sweeps, the aggregation scripts behind every table and figure, the independent re-implementation of Appendix E, and the script that renders the supplementary video; it audited that code against the definitions in this paper, and the corrections, additional control arms and additional seeds that successive revisions adopted were proposed in that audit. Every script was reviewed by the authors, who directed each run; every claim was checked by the authors against the definitions, proofs and measurements already in the paper, and re-worded by them.

Data and result provenance. Every number, table and figure in this paper is computed by the released scripts from recorded simulator runs, public-simulator rollouts, hardware recordings or closed-form definitions, and is traceable to the sources named in the reproducibility statement (Appendix F). No measurement, dataset or plotted value was produced or altered by an LLM, and the experimental protocol and the hyper-parameters are those of the released package's configuration files.

Responsibility. The authors take full responsibility for the entire content of this paper, including every claim, proof and number in it.

A PROOFS AND ADDITIONAL ANALYSIS

A.1 CONSTRAINED BAYES RULE

Fix $X = x$ and write $\mu = \mathbb{E}[Y \mid X = x]$. For any deterministic h,

$$\mathbb{E}[\|Y - h\|^2 \mid X = x] = \mathbb{E}[\|Y - \mu + \mu - h\|^2 \mid X = x] \tag{11}$$

$$= \mathbb{E}[\|Y - \mu\|^2 \mid X = x] + \|\mu - h\|^2, \tag{12}$$

because the cross term has zero conditional expectation. The first term is independent of h. Minimizing over $h \in \mathcal{R}$ is therefore precisely the nearest-point problem, whose unique solution is $\Pi_{\mathcal{R}}(\mu)$. If projection is multi-valued, every nearest point is Bayes optimal and a measurable selection must be specified.

A.2 FINITE-RANK CAPACITY DRAIN

For $X \sim \mathcal{N}(0, I)$,

$$\mathcal{L}_R(W) = \mathbb{E}\|P(M - W)X\|^2 = \|PM - PW\|_F^2. \tag{13}$$

The PTF population problem may be restricted to $PW = W$: with $Q = I - P$, the training loss on projected targets splits as $\mathbb{E}\|(PM - W)X\|^2 = \|PM - PW\|_F^2 + \|QW\|_F^2$ because $PQ = 0$, so removing QW removes a nonnegative term without increasing the rank of W. It therefore reduces to

$$\min_{\mathrm{rank}(W) \le r} \|PM - W\|_F^2, \tag{14}$$

whose solution is $(PM)_r$ by Eckart–Young–Mirsky. Raw population regression returns M_r. Since PM_r has rank at most r, it is feasible for the PTF problem, hence

$$\|PM - (PM)_r\|_F^2 \le \|PM - PM_r\|_F^2. \tag{15}$$

Equality holds if and only if PM_r is itself a best rank-r approximation of PM. This proves Theorem 2.

Affine reachable sets and the intercept. If the reachable set is an affine subspace $r_0 + V$, measure outputs from r_0, so that it becomes the linear subspace V with orthogonal projector P, and let $\mu(X) = y_0 + MX$ with $\mathbb{E}X = 0$. Give both orders a free intercept b, outside the rank budget. Raw regression then returns $b = y_0$ and PTF $b = Py_0$, because $\mathbb{E}X = 0$ separates the constant from the linear part; both deploy $Pb = Py_0$, which is the constant part of $P\mu$, so the constant adds nothing to either side of Eq. 8 and the linear parts are as above. Without an intercept a linear student would have to spend rank on y_0, which is why the theorem is stated for the centred problem.

Minimal strict example. Let $M = \mathrm{diag}(a, b)$, $P = \mathrm{diag}(1, 0)$, $r = 1$, and $0 < a < b$. Raw keeps the unreachable mode b, so $PM_r = 0$ and reachable error is a^2. PTF keeps a, so its reachable error is zero. Scaling b through a creates the sharp rank-one mode transition.

Surplus rank over the reachable dimension does not restore RAW. Take $M = \mathrm{diag}(1, 3, 2)$, $P = \mathrm{diag}(1, 0, 0)$, $r = 2$. Here $\mathrm{rank}(PM) = 1 < r$, so PTF is exact: $(PM)_r = PM$ and $\mathcal{L}_R((PM)_r) = 0$. Raw regression ranks the singular values of M jointly, keeps 3 and 2, and returns $M_r = \mathrm{diag}(0, 3, 2)$, so $PM_r = 0$ and $\mathcal{L}_R(M_r) = \|PM\|_F^2 = 1$. Strict capacity drain therefore persists at a rank strictly greater than the dimension of the reachable subspace. The correct sufficient condition for the two orders to coincide is $r \ge \mathrm{rank}(M)$, which is a statement about the full operator; $r \ge \mathrm{rank}(PM)$ makes PTF exact but says nothing about RAW. The equality condition of Theorem 2 is unaffected: it is PM_r being a best rank-r approximation of PM, and it is the sharp statement.

Sanity check of the linear case. Table 2 sweeps unreachable singular energy and student rank independently, with both orders solved in closed form so that no optimizer, sampling, or seed enters. Take $M = \mathrm{diag}(3, 2, 1, s, s)$ and $P = \mathrm{diag}(1, 1, 1, 0, 0)$, with $Q = I - P$, so that PM has singular values $(3, 2, 1)$, QM has $s \cdot (1, 1)$, $\mathrm{rank}(PM) = 3$ and $\mathrm{rank}(M) = 5$ at the positive scales swept; this is the construction the released script sweeps, and the two component spectra alone would not have fixed $\mathrm{rank}(M)$. Below $s = 1$ an unreachable mode can still enter the top-r truncation — at $r = 4$ it does — but it never displaces a reachable one, and the orders agree everywhere. At $s = 2$ they disagree at $r = 3$ and $r = 4$, that is at and above $\mathrm{rank}(PM)$, and agree at $r = 1$, 2 and at the full-rank control $r = 5$; at $s = 4$, where the unreachable modes outrank every reachable one, they agree only at $r = 5$. Where a reachable and an unreachable singular value tie ($s = 1$ and $s = 2$), the truncation keeps the reachable mode, the convention most favourable to RAW; breaking ties the other way raises the RAW entries to 1 at $s = 1$, $r = 3$–4 and to 5 at $s = 2$, $r = 2$–3, and changes nothing else. The sweep therefore separates the two conditions that the main text distinguishes: PTF's exactness needs $r \ge \mathrm{rank}(PM)$; coincidence of the orders needs $r \ge \mathrm{rank}(M)$, or unreachable modes too small to displace reachable ones.

A.3 SECOND-ORDER PROJECTION-ORDER DISCREPANCY

The expansion is asymptotic in a noise scale, so we state it for a shrinking family. Let $Y_\tau = \mu + \tau\varepsilon$ with $\mathbb{E}[\varepsilon \mid X = x] = 0$, $\mathbb{E}[\|\varepsilon\|^3 \mid X = x] < \infty$, and $\tau \downarrow 0$. Suppose the selected nearest-point

Table 2: Exact population reachable error $\mathcal{L}_R$ for both orders, as a function of student rank r and unreachable-energy scale s. Reachable singular values $(3, 2, 1)$; unreachable $s \cdot (1, 1)$. Closed form, not estimated.

	$s = 0.5$		$s = 1$		$s = 2$		$s = 4$	
r	RAW	PTF	RAW	PTF	RAW	PTF	RAW	PTF
1	5	5	5	5	5	5	14	5
2	1	1	1	1	1	1	14	1
3	0	0	0	0	1	**0**	5	**0**
4	0	0	0	0	1	**0**	1	**0**
5	0	0	0	0	0	0	0	0

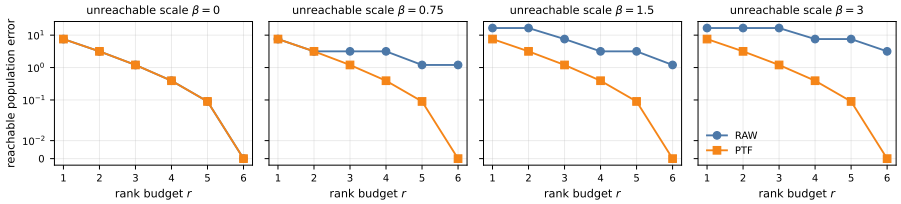


Figure 5: **Exact capacity drain in a linear subspace.** Reachable population error of the best rank-r approximation under each order as the unreachable singular energy grows (scale β); reachable singular values $(3, 2.1, 1.4, 0.9, 0.55, 0.3)$, unreachable $\beta \times (2.8, 2.0, 1.3, 0.85, 0.5, 0.25)$. RAW undergoes discrete mode swaps at small rank; PTF depends only on the reachable operator. Closed form, no stochastic optimizer.

projector is three times continuously differentiable on a neighbourhood N of $\mu(x)$ containing the ball of radius ρ about it, with bounded third derivative there, and that it grows at most linearly, $\|\Pi_{\mathcal{R}}(y)\| \le a + b\|y\|$, which every nearest-point selection onto a nonempty closed set satisfies since $\|\Pi_{\mathcal{R}}(y)\| \le \|y\| + \|y - r_0\|$ for a fixed $r_0 \in \mathcal{R}$. The noise need not be supported in N; Gaussian noise is not. On the event $\{\mu + \tau\varepsilon \in N\}$ the componentwise Taylor expansion gives

$$\Pi_{\mathcal{R}}(\mu + \tau\varepsilon) = \Pi_{\mathcal{R}}(\mu) + \tau J_{\Pi}(\mu)\varepsilon + \tfrac{\tau^2}{2} H_{\Pi}(\mu)[\varepsilon, \varepsilon] + r_3(\tau), \qquad \mathbb{E}\big[\|r_3(\tau)\| \, \mathbf{1}_N\big] = O(\tau^3), \tag{16}$$

and on its complement, whose probability is at most $\tau^3 \mathbb{E}\|\varepsilon\|^3/\rho^3$ by Markov's inequality, the projector's value and each of the three Taylor terms have expectation of order τ^3, because $\mathbb{E}[\|\varepsilon\|^k \mathbf{1}\{\tau\|\varepsilon\| > \rho\}] \le \tau^{3-k}\mathbb{E}\|\varepsilon\|^3/\rho^{3-k}$ for $k = 0, 1, 2$ and the linear growth bounds the projector by $a + b\|\mu\| + b\tau\|\varepsilon\|$. So the expansion holds in expectation with an $O(\tau^3)$ remainder under any noise with a finite conditional third moment. Conditional expectation eliminates the linear term, and contracting each component Hessian with $\Sigma(x) = \mathbb{E}[\varepsilon\varepsilon^\top \mid X = x]$ gives Eq. 9. The formula is local: nearest-point projection onto a nonconvex set can be non-unique or non-smooth beyond the set's reach, and on the piecewise-linear sets of Section 4 the second derivative does not exist at a switching surface, which is why the clamp formula of that section, rather than this expansion, governs that case. Table 10 computes the squared discrepancy on the curved benchmark by Gauss–Hermite quadrature: from $\sigma = 0.05$ to 0.2 it grows by $252\times$, $237\times$ and $132\times$ at $c = 0.5$, 1 and 2 against the $256\times$ of a pure σ^4 law, and from $\sigma = 0.2$ to 0.4 by $15\times$, $13\times$ and $8\times$ against $16\times$, so the expansion is accurate at the two lower curvatures and its remainder is material at $c = 2$ and the largest noise.

Twice differentiability suffices for the $o(\tau^2)$ form. The main text states Eq. 9 for a projector that is only twice differentiable at $\mu = \mu(x)$, and that needs no third moment. Write $R(h) = \Pi_{\mathcal{R}}(\mu+h) - \Pi_{\mathcal{R}}(\mu) - J_{\Pi}(\mu)h - \frac{1}{2}H_{\Pi}(\mu)[h, h]$. Differentiability gives $\|R(h)\| \le \|h\|^2$ for $\|h\| \le \delta$ and some $\delta > 0$, and for every nearest-point selection $\|\Pi_{\mathcal{R}}(\mu+h) - \Pi_{\mathcal{R}}(\mu)\| \le 2\,\mathrm{dist}(\mu, \mathcal{R}) + 2\|h\|$, because the distance function is 1-Lipschitz; so $\|R(h)\| \le K_\delta\|h\|^2$ for $\|h\| > \delta$ as well. Hence $\|R(\tau\varepsilon)\|/\tau^2 \le \max(1, K_\delta)\|\varepsilon\|^2$, which is integrable, and it tends to zero pointwise; dominated convergence gives $\mathbb{E}[R(\tau\varepsilon) \mid X = x] = o(\tau^2)$. Differentiability at $\mu(x)$ in turn needs $\mu(x)$ off the set's medial axis, and holds within the reach of a smooth set. The conditions above buy the sharper $O(\tau^3)$ remainder.

What a deployment projection leaves. An output student that attained m_{PTF} would be deployed as $\Pi_{\mathcal{R}}(m_{\mathrm{PTF}})$. Where $\mathcal{R}$ is a C^2 manifold near $h^\star(x)$, $\Pi_{\mathcal{R}}$ is differentiable at $h^\star(x)$ with Jacobian P_T, the orthogonal projector onto the tangent space there, so

$$\Pi_{\mathcal{R}}(m_{\mathrm{PTF}}(x)) - h^\star(x) = \tfrac{\tau^2}{2}\, P_T\big[H_\Pi(\mu(x)) : \Sigma(x)\big] + o(\tau^2) : \tag{17}$$

the normal part of the discrepancy is projected away, and only its tangential part reaches deployment at this order. By symmetry, isotropic noise about a point on a normal of a circle gives a purely normal discrepancy, which the deployment projection removes at this order; anisotropic, correlated, input-dependent noise, as in the benchmark of Section 5.1, does not. Table 10 gives both parts on both instances.

Bias at a range bound. For one limit $\pi(y) = \min(y, c)$ and $Y = c + \delta + \sigma Z$ with Z standard normal and δ the signed amount by which the conditional mean lies beyond the bound ($\delta > 0$: outside, $\delta < 0$: inside),

$$\mathbb{E}[\pi(Y)] - \pi(\mathbb{E}Y) = \delta_+ - \delta\,\Phi_{\mathcal{N}}(\delta/\sigma) - \sigma\,\varphi_{\mathcal{N}}(\delta/\sigma), \qquad \delta_+ = \max(\delta, 0), \tag{18}$$

with $\Phi_{\mathcal{N}}$ and $\varphi_{\mathcal{N}}$ the standard normal distribution function and density. The bias equals $-\sigma/\sqrt{2\pi}$ at the surface, stays $O(\sigma)$ whenever $|\delta| = O(\sigma)$, and decays exponentially in $|\delta|/\sigma$ once $|\delta| \gg \sigma$ on either side, so at a fixed nonzero distance from the bound it vanishes faster than any power of σ. A deterministic target beyond the bound is clamped on every sample and has zero bias, since then $\mathbb{E}[\pi(Y)] = \pi(\mathbb{E}Y) = c$, and a law symmetric about the centre of a box has zero bias although it crosses the switching surfaces ($Y = \pm 2$ with equal probability gives $\mathbb{E}[\mathrm{clip}(Y, -1, 1)] = 0 = \mathrm{clip}(\mathbb{E}Y, -1, 1)$). The diagnostics the effect calls for are therefore the signed boundary distance of the conditional mean divided by the conditional noise scale, the crossing probability, and $\mathbb{E}[\Pi_{\mathcal{R}}(Y) \mid X] - \Pi_{\mathcal{R}}(\mathbb{E}[Y \mid X])$ itself, all of which need repeated observations per condition; the babble recordings of Appendix C.6 measure how often a fitted limit is binding and not the bias.

A.4 CBD ERROR DECOMPOSITION

Let $g(x)$ be the deployed output, as in Proposition 3. Add and subtract the cached label $\widehat{\Pi}(T_\psi(x))$ and $\Pi_{\mathcal{R}}(T_\psi(x))$, then apply the triangle inequality:

$$\begin{aligned} \|g - h^\star\| &\le \|g - \widehat{\Pi}(T_\psi)\| + \|\widehat{\Pi}(T_\psi) - \Pi_{\mathcal{R}}(T_\psi)\| && (19)\\ &\quad + \|\Pi_{\mathcal{R}}(T_\psi) - \Pi_{\mathcal{R}}(\mu)\| && (20)\\ &\le \epsilon_S + \epsilon_P + L_\Pi \epsilon_T. && (21) \end{aligned}$$

The first term is the deployed student's distance to its own cached label, which is how ϵ_S is defined in the proposition; nothing requires the cached label to lie in $\mathcal{R}$. If the student error is instead measured on the pre-projection regressor, $\|\widehat{f} - \widehat{\Pi}(T_\psi)\| \le \epsilon'_S$, the conversion needs exact deployment, $g = \Pi_{\mathcal{R}}(\widehat{f})$: then $\|g - \Pi_{\mathcal{R}}(\widehat{\Pi}(T_\psi))\| \le L_\Pi \epsilon'_S$ and a further $\|\Pi_{\mathcal{R}}(\widehat{\Pi}(T_\psi)) - \widehat{\Pi}(T_\psi)\| \le \epsilon_P$ appears (it is the distance from $\widehat{\Pi}(T_\psi)$ to $\mathcal{R}$, at most the distance ϵ_P to the point $\Pi_{\mathcal{R}}(T_\psi) \in \mathcal{R}$), so the bound becomes $L_\Pi \epsilon'_S + 2\epsilon_P + L_\Pi \epsilon_T$, and $L_\Pi \epsilon'_S + L_\Pi \epsilon_T$ when $\widehat{\Pi} = \Pi_{\mathcal{R}}$. On the trajectory plant the deployed output is produced by the finite-budget inverse or by executing the emitted commands (Table 3), so ϵ_S is measured on the deployed output directly, as in the proposition, and no conversion is used. The same distance bounds ϵ_S from below, since $g(x) \in \mathcal{R}$: an approximate projector whose labels leave the set puts a floor under ϵ_S that no student capacity removes. Squaring and using $(a + b + c)^2 \le 3(a^2 + b^2 + c^2)$ yields a corresponding MSE bound. The assumption is intentionally local because Euclidean projection is globally nonexpansive only for closed convex sets. For a set of positive reach ρ, such as a compact C^2 manifold, it holds with an explicit constant: on the points within distance $q < \rho$ of $\mathcal{R}$ the nearest point is unique and $\Pi_{\mathcal{R}}$ is $\rho/(\rho - q)$-Lipschitz (Federer, 1959, Theorem 4.8). The proposition therefore applies when $\mu(x)$ and $T_\psi(x)$ lie within the reach, and L_Π grows as they approach it. On the curved benchmark the conditional mean lies at most 0.27 from the parabola (Appendix E), inside its reach $1/c \ge 0.5$ at every tested curvature.

Table 3: What each arm learns and what runs at deployment on the two controlled benchmarks. Π is the exact projector, $\widehat{\Pi}$ the 120-step inverse through the true trajectory plant, T the teacher. The MyoSuite arms are described in Appendix C.7.

Benchmark	Arm	Cached target	Student emits	Deployment	Model access
Curve	RAW / PTF / CBD / oracle CBD	y_i / $\Pi(y_i)$ / $\Pi(T(x_i))$ / $\Pi(\mu(x_i))$	output in $\mathbb{R}^2$	exact Π onto the true set	Π offline (PTF, CBD, oracle) and at deployment
Curve	feasible output	any of the above	curve coordinate u	$r(u)$ on the true set; no inverse	differentiable model $\hat{r}$ during training only
Trajectory	RAW / PTF / CBD / oracle CBD	y_i / $\widehat{\Pi}(y_i)$ / $\widehat{\Pi}(T(x_i))$ / $\widehat{\Pi}(\mu(x_i))$	output trajectory	$\widehat{\Pi}$ through the true plant; 122 plant calls per query batch	true plant offline (PTF, CBD, oracle) and at deployment
Trajectory	feasible output	any of the above	command trajectory	execute on the true plant	differentiable plant copy during training; one call per epoch

B Algorithms

B.1 Target construction

For a training set $\{(x_i, y_i)\}_{i=1}^n$, all methods use the same student class and optimizer. Table 3 states, per experiment, what the student emits and what runs at deployment.

Method	Cached target	Deployment operation
RAW	y_i	project/realize student output
PTF	$\widehat{\Pi}(y_i)$	project/realize student output
CBD	$\widehat{\Pi}(T(x_i))$, T fitted in-sample on all pairs	project/realize student output
CBD (cross-fitted)	$\widehat{\Pi}(T^{-k(i)}(x_i))$	project/realize student output
Oracle CBD	$\Pi(\mu(x_i))$	project/realize student output
Standard distillation	$T(x_i)$	project/realize student output
Partial PTF (α)	$(1-\alpha)\, y_i + \alpha\, \widehat{\Pi}(y_i)$	project/realize student output
Feasible output	any of the above	embedded plant/feasible layer

Here $T^{-k(i)}$ is trained without the fold containing sample i. Cross fitting prevents the teacher from reproducing observation noise in its own training examples. The static and trajectory sweeps report both recipes: the "CBD" rows distil a single teacher of width 128 fitted on all training pairs with patience on the training loss, which is the recipe of the original release, and the "CBD (cross-fitted)" rows distil a five-fold cross-fitted ensemble of the same teacher class; the difference between the two rows is the effect of the teacher protocol and nothing else (Appendix C.2). The teacher sweep, Pendulum and the MyoSuite experiments use cross-fitted teachers throughout. At test time a cross-fitted ensemble may use all folds; only the lightweight student is deployed.

B.2 Iterative trajectory projector

The public simulator uses a bounded cross-entropy method (CEM). Each candidate command sequence is evaluated from an identical snapshot consisting of the environment's state dictionary, the complete MuJoCo integration state of the three model–data pairs the environment holds, the hidden fatigue variables and the target; random-number states are not stored, and the bit-identical replays of the regression test show that the rollouts do not consume any. At every CEM iteration we retain the elite fraction, update the bounded proposal mean and variance, and finally execute the best candidate once from the original snapshot. Caches are keyed by state, target, horizon, plant parameters, optimizer budget, and a projector version.

For the trajectory synthetic, the projector minimizes

$$\frac{1}{T}\sum_{t=1}^{T} \|\widehat{\Phi}(u_{1:t}) - y_t\|^2 + \lambda \frac{1}{T-1}\sum_{t=1}^{T-1} \|u_{t+1} - u_t\|^2, \qquad \lambda = 2 \times 10^{-3}, \tag{22}$$

by 120 Adam steps at learning rate 0.08 from the target itself clipped to the command range, scoring every iterate including the last and keeping, per sample, the iterate that was scored best, and finally

realizing the selected commands once more (122 plant calls per query batch: 121 scored iterates and one realization); the plant saturates commands at ± 1 internally, so the bound is enforced by the plant rather than as a constraint. The plant contains saturation, dead zone, coupling, asymmetric rise/fall response, and history state, and its realization map is continuous on the command box (Appendix C.5). This makes a closed-form pointwise feasible-output parameterization unavailable. The same projector, with the same budget and initialization, realizes every arm's prediction at deployment; the robust variants replace it by an average over five plants whose dead zone and coupling are perturbed by ± 0.06 and $\pm 25\%$.

Because of the command penalty and the finite budget, $\widehat{\Pi}$ is an approximate, regularized nearest point and not the Euclidean projector $\Pi_{\mathcal{R}}$ of the theory; the penalty alone leaves a floor under the ϵ_P of Proposition 3 that more iterations do not remove. The reference against which the trajectory metrics are computed is therefore built separately, per test context, as the best in tracking error to μ of three candidates: the label projector's own output; the same regularized objective run for 480 steps from three initializations (the target and two independent random draws); and an unregularized continuation ($\lambda = 0$, 480 steps) started from the commands the regularized long run found, whose scored iterates include its starting point, so that its tracking error never exceeds the regularized run's. The penalty matters for the search and not only for the labels: on this plant an unregularized run from the target initialization tracks μ *worse* than the regularized one in every test context (0.040 against 0.032 on one seed, at any budget or step size we tried), so an unregularized run from scratch never improves on the label projector; the regularized long run does, and the continuation then removes the penalty's bias from the optimum it found. A zero-command restart is not used, because the gradient with respect to a command inside the dead zone vanishes, so a restart from zero commands never leaves it. We write $\tilde{h}$ rather than $h^\star$ because it remains an approximation; its distance from the label projector's output is recorded for every run (Appendix C.5), and pairwise differences in excess task risk do not depend on it, since its loss is common to every arm; pairwise differences in D_B do. Table 25 also lists $\mathbb{E}\|g - \mu\|^2$, which involves no reference at all.

B.3 Estimating the projector from samples

The analysis is written against an exact Euclidean projector $\Pi_{\mathcal{R}}$, but a deployment system supplies $\mathcal{R}$ only through samples or an offline plant model, so what is applied is an estimate $\widehat{\Pi}$ and the gap between them is the ϵ_P of Proposition 3. Three estimator families cover most practice. A nearest-neighbour projector onto the sample set is feasible by construction but quantized, and its bias is set by sample density. A local tangent projector fits a low-dimensional affine patch and projects onto it, optionally intersected with a coefficient box as in Eq. 27; it is exact when the patch and box describe the set, and it is affine except across the box faces, which is what makes the switching surfaces of Section 4 the operative non-smoothness rather than curvature. An autoencoder projector uses the reconstruction map, which for a denoising autoencoder follows the score of the sample density in the small-noise limit. In this paper the synthetic benchmarks use an exact or deliberately perturbed analytic projector so that ϵ_P is a controlled variable, the MyoSuite experiments use black-box command-space optimization through the simulator, and the physical sets of Appendix C.6 use the box estimator. In every case $\widehat{\Pi}$ is an approximate feasibility operator and not a certificate of the plant's true output set.

B.4 What each method is allowed to know

The orders differ in the information they consume, not only in their loss. RAW sees the raw targets and the deployment projection; PTF additionally calls the offline feasibility operator once per sample; CBD calls it once per teacher prediction and adds a teacher-training cost; a feasible-output student needs a differentiable plant model inside its training loop (the *online* model of Section 5.1) and none at deployment, where it executes its commands. We hold the student class, optimizer, data, transfer set, hyperparameter grid, and evaluation projection fixed within every comparison, and vary plant fidelity deliberately in Section 5.1, where the asymmetry in plant access between an online differentiable model and an offline projector is the experimental question rather than an uncontrolled nuisance.

Compute is not matched across orders, and the reported figures should be read narrowly. Table 4 itemizes the trajectory plant's per-seed cost at width 32: label construction, teacher training, student

training, deployment projection, the total, plant calls at training and at deployment, and cache size. Teacher time is charged only to the arms that use a learned teacher (CBD, its cross-fitted variant, robust CBD and the feasible students on CBD targets). RAW pays only student training and deployment projection; PTF and oracle CBD add the labels; learned CBD adds an in-sample teacher and the cross-fitted variant a five-fold one; the feasible-output students pay one plant call per epoch during training, which is where their student time goes, and one realization of their emitted commands at deployment, a command-network forward pass and one plant call that together take milliseconds (Table 4), whereas every output arm pays the 122-call inverse per deployment query batch (121 scored iterates and one realization; as inverse-search overhead the ledger reads 0 against 121, as forward calls 1 against 122). The warm-started students add a second training stage of the same length without plant calls, and the one on raw targets is charged the PTF inverse it imitates, which no other raw-target arm runs offline; the $2\times$-budget cold students pay twice the plant calls; the compute-matched controls of Table 20 spend 721 plant-batch calls, the raw-target warm student's inverse iterates plus its end-to-end epochs, on restarts or on a plant curriculum; on CBD targets the shared target inverse is charged to every arm on top of that, so there the controls spend more than the warm student rather than the same. Wall-clock times were measured with fifteen shards, fourteen at a time, sharing twenty cores of a machine on which unrelated services used about a sixth of the CPU, and are comparable within the table, not across machines or versions. One-shot and amortized readings differ, because cached targets are reused across widths, seeds, and hyperparameters, so a single sweep's label cost is divided over many students. A comparison that claims a computational advantage needs both a same-data, same-architecture reading and a matched-budget reading; we report the components rather than a single headline number. The main text quotes two readings of this table, and they differ by more than a rounding: counted over every component, the raw-target warm student costs 5.4, 5.2 and 5.0 times RAW at widths 8, 16 and 32; with the deployment column removed the same comparison is 18.0, 16.2 and 14.6, because the deployment inverse is most of RAW's ledger (1.18–1.20 s of 1.69–1.83 s) and costs the feasible student milliseconds.

Table 4: Per-seed cost components on the trajectory plant at width 32 (mean over fifteen seeds). Teacher time is charged only to the arms that use a learned teacher; the label time of the warm-started raw-target student is the PTF inverse it imitates; plant calls are forward calls of the plant, per training run (feasible students, one batch call per epoch) and per deployment query batch (output arms, 121 scored iterates of the inverse and one realization; feasible students, one realization of the emitted commands). A warm-started student's training count is its end-to-end epochs only: the inverse whose commands it imitates is charged to the label column here, and counted in plant-batch calls (121+600) in Table 20. The deployment column is the inverse-search time for the output arms and, for the feasible students (†), the time of their single realization, a command-network forward pass and one plant call on the test batch, which is included in their total. Cache size is the cached target tensor plus, for the warm-started students, the cached inverse commands.

Method	labels (s)	teacher (s)	student (s)	deploy. (s) inverse / real.†	total (s)	plant forward calls train / deploy	cache (MB)
RAW	0.00	0.00	0.63	1.20	1.83	0 / 122	0.000
PTF	1.37	0.00	0.60	1.18	3.15	0 / 122	0.094
CBD	1.34	5.45	0.63	1.21	8.63	0 / 122	0.094
CBD (cross-fitted)	1.42	16.49	0.61	1.17	19.69	0 / 122	0.094
Oracle CBD	1.38	0.00	0.61	1.16	3.16	0 / 122	0.094
Robust CBD	7.05	5.45	0.62	1.13	14.26	0 / 122	0.094
Feasible (RAW targets)	0.00	0.00	7.22	0.002†	7.22	600 / 1	0.000
Feasible (CBD targets)	1.34	5.45	7.31	0.003†	14.11	600 / 1	0.094
Feasible, cold 2× (RAW targets)	0.00	0.00	14.41	0.003†	14.42	1200 / 1	0.000
Feasible, cold 2× (CBD targets)	1.34	5.45	14.62	0.002†	21.42	1200 / 1	0.094
Feasible, warm start (RAW targets)	1.37	0.00	7.83	0.003†	9.20	600 / 1	0.094
Feasible, warm start (CBD targets)	1.34	5.45	7.74	0.003†	14.54	600 / 1	0.188

B.5 CONTROL ARMS

Beyond the three orders, the sweeps carry five controls, all sharing the student class, optimizer, transfer set, evaluation projection, and hyperparameter grid. *Oracle CBD* caches $\Pi_{\mathcal{R}}(\mu(x_i))$ using the analytic conditional mean; it removes ϵ_T from the bound of Proposition 3 and is a target oracle, not an optimization oracle. *CBD (cross-fitted)* replaces the in-sample teacher of the CBD arm by a five-fold cross-fitted ensemble and is otherwise identical. *Standard distillation* caches the same

in-sample teacher's unprojected output $T(x_i)$; comparing it with CBD changes only the projection step, so the difference between them is what feasibility of the label contributes under this recipe, and not a measurement of a term of the bound. *Partial PTF* caches a scalar interpolation between the raw and the projected raw target, $(1-\alpha)y_i + \alpha\widehat{\Pi}(y_i)$ with $\alpha \in \{0.25, 0.5, 0.75\}$, recovering RAW at $\alpha = 0$ and PTF at $\alpha = 1$. *Soft constraint* fits raw targets with an added penalty (weight 10) on the squared distance to $\mathcal{R}$. *Feasible output* embeds a differentiable plant model in the student so that its outputs are feasible by construction; it is trained on each of the four cached targets in turn, and its model carries a controlled parameter error. The latter is the EMBED order of Eq. 4 and is the strongest control wherever its online model is accurate and smooth. *Robust CBD* replaces the projector by an average over a small ensemble of perturbed projectors and is reported as a negative ablation.

C Experimental details

C.1 Two metrics, and exactly how they differ

Two distinct quantities appear in this paper. The *task risk* is the deployed squared loss $\mathcal{L}(g) = \mathbb{E}\|Y - g(X)\|^2$ over feasible predictors $g(X) \in \mathcal{R}$, minimized at $h^\star = \Pi_{\mathcal{R}}(\mu)$. Every synthetic table reports its excess, $\mathcal{L}(g) - \mathcal{L}(h^\star)$, computed on the analytic benchmarks as $\mathbb{E}\|g-\mu\|^2 - \mathbb{E}\|h^\star - \mu\|^2$ so that observation-noise variance does not enter, alongside the *deployed Bayes-target distance*

$$D_B(f) = \mathbb{E}\big\|\Pi_{\mathcal{R}}(f(X)) - h^\star(X)\big\|^2, \tag{23}$$

which measures how closely a training order recovers the constrained Bayes predictor. The MyoSuite results in Table 32 report task-side tracking MSE and are unaffected by the distinction below.

D_B is not excess task risk. For every feasible g,

$$\mathcal{L}(g) - \mathcal{L}(h^\star) = \mathbb{E}\|g - h^\star\|^2 + 2\,\mathbb{E}\big\langle g - h^\star,\ h^\star - \mu\big\rangle. \tag{24}$$

The cross term vanishes when $\mathcal{R}$ is an affine subspace with its orthogonal projector; it is nonnegative when $\mathcal{R}$ is convex, by the variational inequality characterizing Euclidean projection; and it can take either sign on a nonconvex set. So on convex reachable sets D_B is a lower bound on excess task risk, and the two criteria can order methods differently.

Where the two agree, and by how much. If $\mathcal{R}$ is a C^2 manifold and $g, h^\star$ lie close together on it, $g - h^\star$ is tangent to first order while $h^\star - \mu$ is normal, so the cross term is $\pm\kappa\|g - h^\star\|^2\|h^\star - \mu\|$ to leading order, with κ the normal curvature and the sign fixed by which side of the surface μ lies on. This is second order in the output displacement $g - h^\star$, the same order as D_B itself, so it is *not* a higher-order correction: the two criteria agree to leading order only when the product $\kappa\|h^\star - \mu\|$ of curvature and mean-to-set distance is small. The unit circle makes the point exactly. With $\mathcal{R} = S^1$, $\mu = \rho e_1$ for $\rho > 1$, $h^\star = e_1$ and $g = (\cos\theta, \sin\theta)$, one has $D_B = 2(1 - \cos\theta)$ and $2\langle g - h^\star, h^\star - \mu\rangle = (\rho - 1)D_B$, so the excess risk is ρD_B for every θ, however small. The curved benchmark of Section 5.1 has $\kappa\|h^\star - \mu\|$ of order 0.1, and the measurement bears the formula out: over the rows of the plain arms the cross term is negative in every one of the $5{,}940$ rows, because μ lies on the concave side, and is a stable fraction of D_B: 90% of rows lie between -18% and -4% (median -6.5%, extremes -36% and -2%), so D_B slightly overstates excess risk there and the winner maps under the two criteria agree in every cell. A strictly violated box bound, by contrast, gives a correction that is *first* order in the displacement, as the next proposition shows.

Where they differ, exactly. On the box geometry that actuator limits produce, the gap is first order and has a closed form.

Proposition 4 (Metric gap at an active bound)**.** *Let $\mathcal{R} = \{c + Uz :\ \ell \le z \le u\}$ with $U^\top U = I$, and for a context x write $m = U^\top(\mu(x) - c)$, $\xi = \mathrm{clip}(m, \ell, u)$, and $\gamma = U^\top(g(x) - c)$ for any feasible g; m, ξ, γ and the active set $A = \{j : \xi_j \neq m_j\}$ all depend on x. Then $h^\star = c + U\xi$ and, with $\mathcal{L}_x(g) = \mathbb{E}[\|Y - g(x)\|^2 \mid X = x]$ the conditional risk,*

$$\mathcal{L}_x(g) - \mathcal{L}_x(h^\star) = \textstyle\sum_j(\gamma_j - \xi_j)^2 \ +\ 2\sum_{j\in A}\big|\gamma_j - \xi_j\big|\,\delta_j, \qquad \delta_j = |m_j - \xi_j|, \tag{25}$$

so that $\mathcal{L}(g) - \mathcal{L}(h^\star) = D_B + 2\,\mathbb{E}_X\big[\sum_{j\in A(X)}|\gamma_j - \xi_j|\,\delta_j\big]$ with $D_B = \mathbb{E}_X\sum_j(\gamma_j - \xi_j)^2$; so each active coordinate adds a term proportional to how far the conditional mean lies beyond its bound.

Table 5: **Contrasts declared before the fifteen-seed run**, on all seeds and on the ten seeds (5–14) collected after the contrasts were selected from the five-seed pilot. Negative means the first arm is better; within-cell contrasts use the paired standard error over seeds, the pooled contrast the seed-block standard error; a check marks $|t| > t_{0.975,n-1}$. Four tests in three families, no familywise claim.

	all seeds ($n = 15$, $t_{0.975} = 2.14$)		seeds after selection ($n = 10$, $t_{0.975} = 2.26$)	
Contrast (first − second)	mean ± s.e.	$\lvert t\rvert$	mean ± s.e.	$\lvert t\rvert$
excess task risk				
CBD − PTF, $c = 2$, $\sigma = 0.4$, w16	$-5.98 \times 10^{-4} \pm 1.2 \times 10^{-4}$	4.9✓	$-6.81 \times 10^{-4} \pm 1.6 \times 10^{-4}$	4.4✓
CBD − PTF, $c = 2$, $\sigma = 0.4$, w32	$-6.73 \times 10^{-4} \pm 1.3 \times 10^{-4}$	5.3✓	$-7.67 \times 10^{-4} \pm 1.7 \times 10^{-4}$	4.6✓
RAW − CBD, $c = 1$, $\sigma = 0.05$, w32	$-3.19 \times 10^{-5} \pm 2.5 \times 10^{-6}$	12.8✓	$-3.30 \times 10^{-5} \pm 3.7 \times 10^{-6}$	8.9✓
feasible (RAW) − CBD, pooled	$-1.35 \times 10^{-4} \pm 1.2 \times 10^{-5}$	11.2✓	$-1.39 \times 10^{-4} \pm 1.7 \times 10^{-5}$	8.0✓
D_B				
CBD − PTF, $c = 2$, $\sigma = 0.4$, w16	$-1.03 \times 10^{-3} \pm 1.8 \times 10^{-4}$	5.6✓	$-1.15 \times 10^{-3} \pm 2.3 \times 10^{-4}$	5.0✓
CBD − PTF, $c = 2$, $\sigma = 0.4$, w32	$-1.13 \times 10^{-3} \pm 2.0 \times 10^{-4}$	5.6✓	$-1.27 \times 10^{-3} \pm 2.6 \times 10^{-4}$	4.9✓
RAW − CBD, $c = 1$, $\sigma = 0.05$, w32	$-3.30 \times 10^{-5} \pm 2.8 \times 10^{-6}$	11.7✓	$-3.41 \times 10^{-5} \pm 4.2 \times 10^{-6}$	8.1✓
feasible (RAW) − CBD, pooled	$-1.44 \times 10^{-4} \pm 1.3 \times 10^{-5}$	11.3✓	$-1.50 \times 10^{-4} \pm 1.9 \times 10^{-5}$	8.1✓

Proof. Fix x and write $\mathcal{L}_x(g) = \mathbb{E}[\|Y - g(x)\|^2 \mid X{=}x]$ for the conditional risk, so that $\mathcal{L}(g) = \mathbb{E}_X[\mathcal{L}_X(g)]$. $g(x)$ and $h^\star(x)$ lie in the same affine subspace, so the component of Y orthogonal to it contributes equally to both conditional risks and cancels. Writing $\eta = U^\top(Y - c)$ and using $\mathbb{E}[\eta \mid X{=}x] = m$, $\mathcal{L}_x(g) - \mathcal{L}_x(h^\star) = \|\gamma - m\|^2 - \|\xi - m\|^2 = \|\gamma - \xi\|^2 + 2\langle\gamma - \xi, \xi - m\rangle$. For $j \notin A$ the summand vanishes. For $j \in A$ at an upper bound, $\xi_j = u_j < m_j$ and $\gamma_j \le u_j$, so $(\gamma_j - \xi_j)(\xi_j - m_j) = |\gamma_j - \xi_j|\,\delta_j \ge 0$; the lower-bound case is symmetric. Eq. 25 is this identity at the fixed context; taking $\mathbb{E}_X$ of both sides, with γ, ξ, m, A and δ_j all functions of X, gives the population statement. □

Two consequences matter for reading the tables. The gap can arise *only* on coordinates whose bound the *conditional mean* violates, so it is invisible in cells where the conditional mean stays inside the box; an active bound is necessary but not sufficient, since each term carries the factor $|\gamma_j - \xi_j|$ and the gap still vanishes for a predictor that matches the Bayes output on those coordinates while erring elsewhere (g_A below is such a predictor). On the measured hardware of Appendix C.6 the fitted limits are binding for 86–93% of held-out target frames, which is the analogous statement about individual targets and is a precondition, not a measurement of δ_j. And the gap grows with a method's error *at those coordinates*, so a training order that is biased toward the interior where the target saturates pays it in full. Project-then-fit has precisely that bias (Section 4). The trajectory plant of Section 5.2 saturates without being a box, with coupled commands and a history state, so the proposition is its idealized archetype rather than an identity there; the two criteria do separate on it (Table 26), and Table 30 measures the cross term directly.

A ranking reversal. The two criteria genuinely disagree. Take $\mathcal{R} = [-1, 1]^2$, deterministic $Y = \mu = (0, 2)$, so $h^\star = (0, 1)$ and $\mathcal{L}(h^\star) = 1$. For $g_A = (0.9, 1)$ and $g_B = (0, 0.5)$:

Feasible output	D_B	Excess task risk
$g_A = (0.9, 1)$	0.81	0.81
$g_B = (0, 0.5)$	**0.25**	1.25

D_B prefers g_B; the task objective prefers g_A. The difference is the missing normal-direction penalty of Eq. 25.

Reporting. Every row of every synthetic sweep carries both quantities, computed from the same deployed prediction, so their difference is the cross term of Eq. 24 on the fitted model itself. The main-text figures show excess task risk and the appendix repeats each display under D_B; where a single number is quoted in prose it is the excess task risk unless labelled D_B.

Uncertainty: the seed is the replication unit. We retain seed-level rows. Within one cell of a sweep the seeds are independent, and a comparison between two arms is the paired difference over

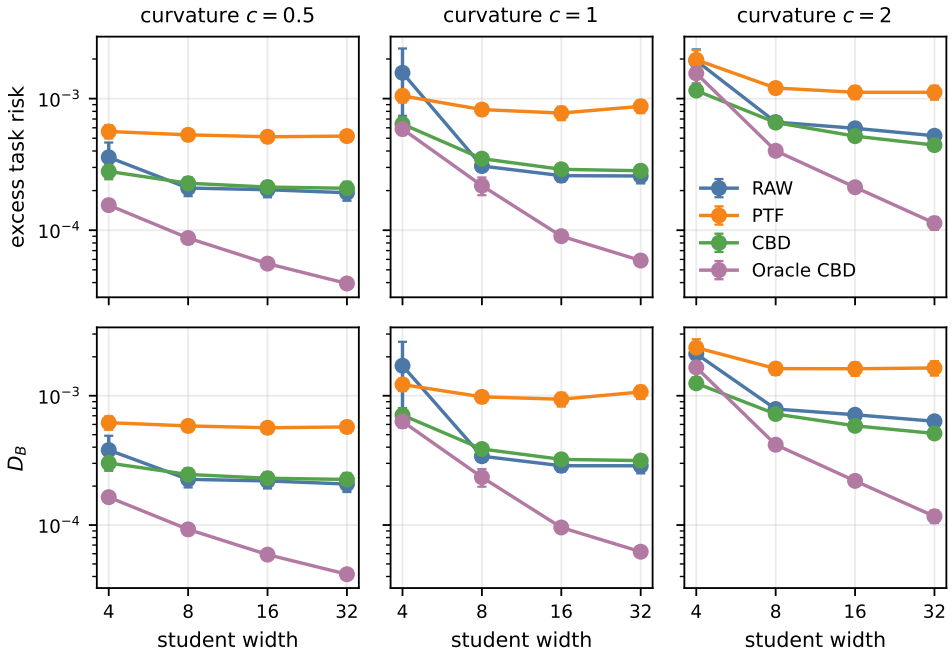
seeds with its standard error over seeds. Across cells they are not: the static driver re-seeds the generator to the same value at the top of every (curvature, noise) cell of a seed, so all 36 cells of one seed share their training contexts and their base Gaussian draws, and the $36 \times n$ paired differences of a pooled contrast are not $36 \times n$ independent observations. A pooled contrast is therefore aggregated by *seed block*: the 36 cell differences are averaged within each seed, and the standard error is taken over the n seed averages. On the five-seed pilot the seed-block standard errors were 1.5 to 2.6 times the pooled ones, which is why the sweeps use fifteen seeds. A comparison is called *decided* when its paired mean exceeds $t_{0.975,n-1}$ standard errors, a two-sided 95% paired t-test: 2.145 for the fifteen-seed sweeps and 2.776 for the five-seed ones. The winner maps hatch every cell whose top-two comparison is not decided, and they remain exploratory summaries of best means over many cells, with no correction for the number of cells. Before the fifteen-seed run we predeclared three families of contrasts, four tests in all, chosen from the five-seed run: CBD against PTF at $c = 2$, $\sigma = 0.4$, widths 16 and 32 (the regime in which CBD is predicted to help); RAW against CBD at $c = 1$, $\sigma = 0.05$, width 32 (a RAW-favourable control); and the pooled feasible-output-against-CBD contrast. Their outcomes are reported with the static results and in Table 5, which gives each contrast twice: on all fifteen seeds, and on the ten seeds (5–14) that were run only after the contrasts had been selected, since the five pilot seeds cannot confirm a hypothesis they suggested. These are four individual tests in three families (two CBD-versus-PTF cells, one control, one pooled contrast); no multiplicity correction is applied and no familywise claim is made. We write "we did not resolve a difference" rather than "tied" whenever a contrast is not decided, since no equivalence margin was specified. Per-method summaries in this paper are mean $\pm$ standard deviation; paired contrasts, in the text and in the paired tables, are the mean difference $\pm$ its standard error over seeds; figure error bars are stated in their captions. Tables 8, 26 and 34 are the paired tables.

C.2 Static nonlinear sweep

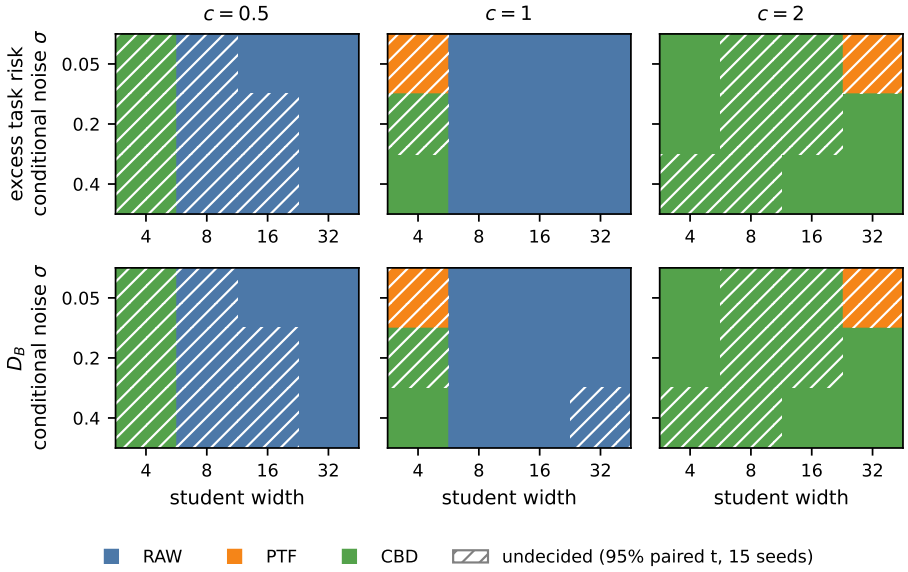
Table 6: Controlled static grid. Patience is on the training loss and is a convergence criterion, not held-out early stopping.

Control	Values
Seeds	0–14 (fifteen)
Training / test contexts	2,048 / 512
Student width	4, 8, 16, 32 (depth 2)
Student optimizer	Adam, lr 5×10^{-3}, 700 epochs, patience 100 (training loss)
Checkpoint	the iterate that produced the best training loss
Teacher (CBD, KD)	width 128, depth 3, in-sample on all pairs, 1,200 epochs, lr 3×10^{-3}
Teacher (cross-fitted CBD)	same class, five folds, one model per fold
Reachable curvature c	0.5, 1.0, 2.0
Conditional noise std. σ	0.05, 0.2, 0.4
Normal offset scale	0.3
$\mathbb{E}[\Pi_{\mathcal{R}}(Y) \mid X]$	64×64 Gauss–Hermite nodes per test context
Feasible-layer curvature error	$-0.5, -0.25, 0, 0.25, 0.5$ (relative)
Partial PTF α	0.25, 0.5, 0.75
Soft-constraint weight	10
Learned targets	RAW, PTF, CBD, CBD (cross-fitted), partial PTF, standard distillation
Controls	oracle CBD, four feasible-output variants, soft constraint
Seed-level rows	16,200

Winner counts. Table 7 summarizes the maps of Figure 6 under both criteria and under three method families. The three-order and four-arm best-mean maps are identical under the two criteria (the full three-order ranking is the same in every cell; the full four-arm ranking differs in two cells, and the full ranking of the five arms of Figure 3a in four, with first place exchanged in one, $c = 2$, $\sigma = 0.4$, width 8, between the exact feasible student and oracle CBD), and their decided cells differ in one ($c = 1$, $\sigma = 0.4$, width 32: RAW against CBD at $|t| = 2.29$ under task risk and 2.08 under D_B, so 19 against 18 decided cells); the map over all learned arms differs in two cells, where a feasible-output student and standard distillation swap places (24 against 22 feasible cells and 6 against 8 distillation cells). Among the four target-based arms, oracle CBD has the best mean in 20 of 36 cells, RAW and learned CBD in 7 each and PTF in 2; replacing the in-sample CBD arm by the



(a) Fifteen-seed means at conditional noise $\sigma = 0.4$, excess task risk (top) and D_B (bottom); error bars are standard errors over seeds.



(b) Best mean among learned RAW, PTF and CBD over all 36 cells under each criterion, an exploratory summary. Hatched cells are not decided: the paired difference between the two best arms fails a two-sided 95% paired t-test over the fifteen seeds.

Figure 6: **Capacity–curvature–noise regimes under both criteria**; Figure 3 shows the excess-task-risk rows.

cross-fitted one leaves the three-order map unchanged; and among all learned arms with an exact online model, the feasible-output students take 24 cells between them under excess task risk, the RAW-target one 15; oracle CBD has a lower mean than that student in 19 of the 36 cells. Across the three noise levels the best learned order changes in two (curvature, width) combinations, $c = 1$ at width 4 and $c = 2$ at width 32, neither decided.

Table 7: Cells (of 36) in which each arm has the best fifteen-seed mean, the number of cells whose top-two comparison is decided by a 95% paired t-test (in parentheses, by winner), the same three-order map with the cross-fitted CBD arm in place of the in-sample one, and the maps over four target-based arms and over all learned arms with an exact online model.

Criterion	RAW/PTF/CBD	decided (by winner)	with cross-fitted CBD	RAW/PTF/CBD/oracle	feasible/RAW/KD/CBD/PTF
excess task risk	18/2/16	19 (13/0/6)	18/2/16	7/2/7/20	24/4/6/1/1
D_B	18/2/16	18 (12/0/6)	18/2/16	7/2/7/20	22/4/8/1/1

Paired differences. Table 8 gives the pooled paired differences of every static comparison under both criteria; the contrasts declared before the fifteen-seed run are in Table 5 (Appendix C.1).

Table 8: Paired differences on the static grid, pooled over the 36 cells, for both criteria. Negative means the first arm is better. The standard error is the seed-block standard error over the fifteen seeds (the 36 cells of a seed are averaged first, because they share their draws), $|t|$ is the ratio, and a check marks a comparison decided at the two-sided 95% level ($|t| > 2.14$); "better" counts the 540 cell-seed pairs in which the first arm is better. The three indented rows pool CBD minus PTF within one curvature (12 cells, 180 cell-seed pairs each). Feasible-output rows use the exact online model (curvature error 0).

	excess task risk			D_B		
Comparison	mean ± s.e.	$\|t\|$	better	mean ± s.e.	$\|t\|$	better
PTF − RAW	$+6.95 \times 10^{-5} \pm 7.1 \times 10^{-5}$	1.0	138/540	$+1.16 \times 10^{-4} \pm 7.9 \times 10^{-5}$	1.5	138/540
CBD − RAW	$-1.36 \times 10^{-4} \pm 6.7 \times 10^{-5}$	2.0	234/540	$-1.58 \times 10^{-4} \pm 7.4 \times 10^{-5}$	2.1	240/540
CBD − PTF	$-2.06 \times 10^{-4} \pm 4.3 \times 10^{-5}$	4.7✓	373/540	$-2.74 \times 10^{-4} \pm 5.3 \times 10^{-5}$	5.2✓	376/540
pooled within $c = 0.5$	$-1.03 \times 10^{-4} \pm 2.1 \times 10^{-5}$	5.0✓	115/180	$-1.16 \times 10^{-4} \pm 2.3 \times 10^{-5}$	5.0✓	115/180
pooled within $c = 1$	$-1.74 \times 10^{-4} \pm 4.4 \times 10^{-5}$	4.0✓	126/180	$-2.20 \times 10^{-4} \pm 5.2 \times 10^{-5}$	4.3✓	128/180
pooled within $c = 2$	$-3.39 \times 10^{-4} \pm 8.5 \times 10^{-5}$	4.0✓	132/180	$-4.87 \times 10^{-4} \pm 1.0 \times 10^{-4}$	4.7✓	133/180
Oracle CBD − CBD	$-2.12 \times 10^{-5} \pm 2.6 \times 10^{-5}$	0.8	365/540	$-2.96 \times 10^{-5} \pm 2.8 \times 10^{-5}$	1.0	370/540
CBD (cross-fitted) − CBD	$+2.90 \times 10^{-5} \pm 2.1 \times 10^{-5}$	1.4	271/540	$+3.12 \times 10^{-5} \pm 2.3 \times 10^{-5}$	1.4	270/540
Standard distillation − CBD	$-3.94 \times 10^{-6} \pm 2.8 \times 10^{-5}$	0.1	322/540	$+6.23 \times 10^{-6} \pm 3.1 \times 10^{-5}$	0.2	315/540
Soft constraint − RAW	$+1.68 \times 10^{-3} \pm 1.1 \times 10^{-3}$	1.5	112/540	$+1.84 \times 10^{-3} \pm 1.3 \times 10^{-3}$	1.4	140/540
Feasible (RAW targets) − RAW	$-2.71 \times 10^{-4} \pm 7.3 \times 10^{-5}$	3.7✓	367/540	$-3.03 \times 10^{-4} \pm 8.2 \times 10^{-5}$	3.7✓	365/540
Feasible (RAW targets) − CBD	$-1.35 \times 10^{-4} \pm 1.2 \times 10^{-5}$	11.2✓	439/540	$-1.44 \times 10^{-4} \pm 1.3 \times 10^{-5}$	11.3✓	436/540
Feasible (CBD targets) − CBD	$-1.32 \times 10^{-4} \pm 1.3 \times 10^{-5}$	10.4✓	442/540	$-1.42 \times 10^{-4} \pm 1.4 \times 10^{-5}$	10.5✓	437/540
Feasible (PTF targets) − PTF	$-1.92 \times 10^{-4} \pm 2.5 \times 10^{-5}$	7.7✓	426/540	$-2.30 \times 10^{-4} \pm 2.7 \times 10^{-5}$	8.4✓	431/540

Teacher protocol. Table 9 isolates the effect of cross-fitting the teacher: the cross-fitted CBD arm differs from the CBD arm only in the teacher, and the teacher conditional-mean errors of the two protocols are listed per cell. For the cross-fitted teacher two errors are given. The ensemble-mean error is that of the average of the five fold models on the test contexts; the single-fold error is the average over the five fold models of each model's own test error, and it is the matched diagnostic, because every cached label of the cross-fitted arm is produced by one held-out fold model and not by the ensemble. The single-fold error exceeds the ensemble's by the variance the averaging removes, and it is the number a teacher-error bound would take.

The Monte-Carlo floor of the earlier Table 10. The original release estimated $\mathbb{E}[\Pi_{\mathcal{R}}(Y) \mid X]$ by a 256-draw Monte-Carlo mean and reported its squared distance to $\Pi_{\mathcal{R}}(\mu)$. That plug-in estimator is biased upward by $\operatorname{tr} \operatorname{Cov}(\Pi_{\mathcal{R}}(Y) \mid X)/(Nd)$, a floor that repeating seeds does not remove. Table 11 compares it with the quadrature value now used: at $\sigma = 0.05$ the floor was 54–95% of the number previously reported, and the earlier statement that the discrepancy grew only 7× per doubling of the noise, from which we had inferred a material higher-order remainder at every curvature, was an artefact of that floor.

Table 9: Cross-fitted minus in-sample CBD on the static grid (pooled, seed-block standard error, fifteen seeds), the number of cells in which the cross-fitted arm has the lower mean, and the teachers' conditional-mean MSE per curvature and noise: the in-sample teacher, the cross-fitted ensemble mean, and the average single fold model (the matched diagnostic), with the ratio of the last two.

Cross-fitted − in-sample CBD	mean ± s.e. (seed)	$\lvert t\rvert$	better	cells
excess task risk	$+2.90 \times 10^{-5} \pm 2.1 \times 10^{-5}$	1.37	271/540	17/36
D_B	$+3.12 \times 10^{-5} \pm 2.3 \times 10^{-5}$	1.35	270/540	17/36

teacher conditional-mean MSE	in-sample	cross-fitted, ensemble mean	cross-fitted, single fold	fold / ensemble
$c = 0.5, \sigma = 0.05$	2.44×10^{-5}	2.16×10^{-5}	2.55×10^{-5}	1.18
$c = 0.5, \sigma = 0.2$	1.47×10^{-4}	1.38×10^{-4}	1.71×10^{-4}	1.24
$c = 0.5, \sigma = 0.4$	6.12×10^{-4}	6.43×10^{-4}	7.94×10^{-4}	1.23
$c = 1, \sigma = 0.05$	3.83×10^{-5}	3.77×10^{-5}	3.99×10^{-5}	1.06
$c = 1, \sigma = 0.2$	1.64×10^{-4}	1.65×10^{-4}	1.93×10^{-4}	1.17
$c = 1, \sigma = 0.4$	6.41×10^{-4}	6.79×10^{-4}	8.19×10^{-4}	1.21
$c = 2, \sigma = 0.05$	5.34×10^{-5}	5.45×10^{-5}	6.28×10^{-5}	1.15
$c = 2, \sigma = 0.2$	2.27×10^{-4}	2.16×10^{-4}	2.54×10^{-4}	1.18
$c = 2, \sigma = 0.4$	7.55×10^{-4}	7.43×10^{-4}	8.96×10^{-4}	1.21

Table 10: **Population projection-order discrepancy** $\mathbb{E}\|\mathbb{E}[\Pi_{\mathcal{R}}(Y) \mid X] - \Pi_{\mathcal{R}}(\mathbb{E}[Y \mid X])\|^2$ (per-coordinate mean over the 512 test contexts) on the curved benchmark of Section 5.1 and on the independent re-implementation of Appendix E, computed by 64×64-node Gauss–Hermite quadrature over the two standard normals that generate the conditional noise; the values are numerical, and differ from a 128×128-node integration by at most 0.4% (benchmark) and 0.7% (re-implementation) over the cells shown. Both instances are shown at the same numeric curvature c (the re-implementation's $\kappa = c/2$). Rows: curvature c, each with the *ambient* discrepancy above and the *deployed* one, $\mathbb{E}\|\Pi_{\mathcal{R}}(\mathbb{E}[\Pi_{\mathcal{R}}(Y) \mid X]) - \Pi_{\mathcal{R}}(\mathbb{E}[Y \mid X])\|^2$, the Bayes-target distance of an unrestricted population PTF student after exact deployment projection; columns: conditional noise σ. Both are population quantities, independent of any student, and neither is an excess task risk.

		released benchmark			re-implementation at the same c ($\kappa = c/2$)		
c		$\sigma = 0.05$	$\sigma = 0.2$	$\sigma = 0.4$	$\sigma = 0.05$	$\sigma = 0.2$	$\sigma = 0.4$
0.5	ambient	1.84×10^{-7}	4.63×10^{-5}	7.06×10^{-4}	2.35×10^{-7}	6.14×10^{-5}	1.05×10^{-3}
	deployed	9.72×10^{-8}	2.45×10^{-5}	3.69×10^{-4}	8.82×10^{-9}	3.42×10^{-6}	1.30×10^{-4}
1	ambient	6.45×10^{-7}	1.53×10^{-4}	2.03×10^{-3}	1.03×10^{-6}	2.61×10^{-4}	3.67×10^{-3}
	deployed	3.45×10^{-7}	7.90×10^{-5}	9.25×10^{-4}	1.78×10^{-7}	3.98×10^{-5}	3.56×10^{-4}
2	ambient	5.00×10^{-6}	6.59×10^{-4}	5.19×10^{-3}	1.66×10^{-5}	1.33×10^{-3}	8.72×10^{-3}
	deployed	2.90×10^{-6}	2.89×10^{-4}	1.87×10^{-3}	7.71×10^{-6}	3.04×10^{-4}	7.28×10^{-4}

Table 11: Released benchmark: projection-order discrepancy by 64-node quadrature, the Monte-Carlo floor of the 256-draw plug-in estimator, the plug-in value realized over five seeds, and the floor's share of it.

c	σ	exact (quadrature)	floor $\operatorname{tr}\operatorname{Cov}/(Nd)$	plug-in, $N = 256$	floor share
0.5	0.05	1.84×10^{-7}	3.58×10^{-6}	3.91×10^{-6}	95%
0.5	0.2	4.63×10^{-5}	5.75×10^{-5}	1.04×10^{-4}	55%
0.5	0.4	7.06×10^{-4}	2.33×10^{-4}	9.36×10^{-4}	25%
1	0.05	6.45×10^{-7}	4.53×10^{-6}	5.24×10^{-6}	88%
1	0.2	1.53×10^{-4}	7.30×10^{-5}	2.22×10^{-4}	32%
1	0.4	2.03×10^{-3}	2.98×10^{-4}	2.31×10^{-3}	13%
2	0.05	5.00×10^{-6}	5.89×10^{-6}	1.04×10^{-5}	54%
2	0.2	6.59×10^{-4}	9.43×10^{-5}	7.39×10^{-4}	13%
2	0.4	5.19×10^{-3}	3.62×10^{-4}	5.51×10^{-3}	7%

Representative cells. Table 12 gives means at the highest noise for every curvature and width under both criteria. Oracle CBD supplies an oracle *target*; its finite neural student still has optimization and approximation error. Consequently it need not have the best mean in every small-width cell, which is visible at $c = 2$, width 4, where learned CBD and the feasible student both beat it.

Table 12: Static means at $\sigma = 0.4$ over fifteen seeds (lower is better; bold is the row minimum). KD is standard distillation; the feasible column uses the exact online model and RAW targets.

c	width	RAW	PTF	CBD	CBD (xfit)	oracle CBD	KD	feasible
excess task risk								
0.5	4	3.58×10^{-4}	5.63×10^{-4}	2.80×10^{-4}	2.74×10^{-4}	$\mathbf{1.55 \times 10^{-4}}$	3.36×10^{-4}	2.09×10^{-4}
0.5	8	2.09×10^{-4}	5.31×10^{-4}	2.28×10^{-4}	2.45×10^{-4}	$\mathbf{8.72 \times 10^{-5}}$	2.09×10^{-4}	2.00×10^{-4}
0.5	16	2.03×10^{-4}	5.13×10^{-4}	2.13×10^{-4}	2.06×10^{-4}	$\mathbf{5.57 \times 10^{-5}}$	2.00×10^{-4}	2.06×10^{-4}
0.5	32	1.93×10^{-4}	5.19×10^{-4}	2.08×10^{-4}	2.05×10^{-4}	$\mathbf{3.94 \times 10^{-5}}$	1.92×10^{-4}	1.92×10^{-4}
1	4	1.57×10^{-3}	1.05×10^{-3}	6.41×10^{-4}	5.03×10^{-4}	5.88×10^{-4}	5.53×10^{-4}	$\mathbf{2.83 \times 10^{-4}}$
1	8	3.07×10^{-4}	8.26×10^{-4}	3.50×10^{-4}	3.86×10^{-4}	$\mathbf{2.18 \times 10^{-4}}$	2.68×10^{-4}	2.71×10^{-4}
1	16	2.60×10^{-4}	7.77×10^{-4}	2.90×10^{-4}	2.96×10^{-4}	$\mathbf{9.01 \times 10^{-5}}$	2.45×10^{-4}	2.46×10^{-4}
1	32	2.58×10^{-4}	8.74×10^{-4}	2.83×10^{-4}	2.89×10^{-4}	$\mathbf{5.88 \times 10^{-5}}$	2.50×10^{-4}	2.56×10^{-4}
2	4	1.94×10^{-3}	1.97×10^{-3}	1.15×10^{-3}	1.68×10^{-3}	1.56×10^{-3}	1.14×10^{-3}	$\mathbf{4.51 \times 10^{-4}}$
2	8	6.63×10^{-4}	1.20×10^{-3}	6.58×10^{-4}	6.37×10^{-4}	4.01×10^{-4}	7.13×10^{-4}	$\mathbf{3.97 \times 10^{-4}}$
2	16	5.95×10^{-4}	1.12×10^{-3}	5.18×10^{-4}	4.96×10^{-4}	$\mathbf{2.12 \times 10^{-4}}$	5.75×10^{-4}	3.78×10^{-4}
2	32	5.22×10^{-4}	1.12×10^{-3}	4.44×10^{-4}	4.39×10^{-4}	$\mathbf{1.13 \times 10^{-4}}$	5.41×10^{-4}	3.69×10^{-4}
D_B								
0.5	4	3.80×10^{-4}	6.19×10^{-4}	3.02×10^{-4}	2.95×10^{-4}	$\mathbf{1.64 \times 10^{-4}}$	3.59×10^{-4}	2.26×10^{-4}
0.5	8	2.26×10^{-4}	5.84×10^{-4}	2.46×10^{-4}	2.64×10^{-4}	$\mathbf{9.26 \times 10^{-5}}$	2.25×10^{-4}	2.16×10^{-4}
0.5	16	2.19×10^{-4}	5.66×10^{-4}	2.30×10^{-4}	2.23×10^{-4}	$\mathbf{5.90 \times 10^{-5}}$	2.15×10^{-4}	2.22×10^{-4}
0.5	32	2.08×10^{-4}	5.73×10^{-4}	2.25×10^{-4}	2.22×10^{-4}	$\mathbf{4.16 \times 10^{-5}}$	2.07×10^{-4}	2.08×10^{-4}
1	4	1.71×10^{-3}	1.22×10^{-3}	7.08×10^{-4}	5.57×10^{-4}	6.33×10^{-4}	6.02×10^{-4}	$\mathbf{3.15 \times 10^{-4}}$
1	8	3.41×10^{-4}	9.80×10^{-4}	3.87×10^{-4}	4.24×10^{-4}	$\mathbf{2.34 \times 10^{-4}}$	2.94×10^{-4}	3.02×10^{-4}
1	16	2.88×10^{-4}	9.40×10^{-4}	3.22×10^{-4}	3.28×10^{-4}	$\mathbf{9.57 \times 10^{-5}}$	2.72×10^{-4}	2.73×10^{-4}
1	32	2.87×10^{-4}	1.07×10^{-3}	3.15×10^{-4}	3.21×10^{-4}	$\mathbf{6.22 \times 10^{-5}}$	2.79×10^{-4}	2.85×10^{-4}
2	4	2.11×10^{-3}	2.36×10^{-3}	1.25×10^{-3}	1.83×10^{-3}	1.66×10^{-3}	1.24×10^{-3}	$\mathbf{4.98 \times 10^{-4}}$
2	8	7.89×10^{-4}	1.62×10^{-3}	7.22×10^{-4}	6.95×10^{-4}	$\mathbf{4.18 \times 10^{-4}}$	8.25×10^{-4}	4.29×10^{-4}
2	16	7.13×10^{-4}	1.62×10^{-3}	5.85×10^{-4}	5.54×10^{-4}	$\mathbf{2.19 \times 10^{-4}}$	6.98×10^{-4}	4.15×10^{-4}
2	32	6.35×10^{-4}	1.64×10^{-3}	5.11×10^{-4}	5.03×10^{-4}	$\mathbf{1.17 \times 10^{-4}}$	6.59×10^{-4}	4.12×10^{-4}

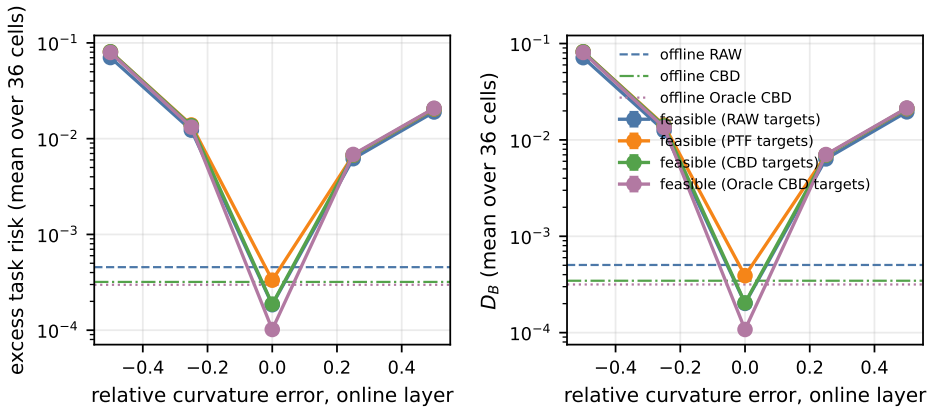


Figure 7: **Feasible-output students against the error of the model they train through** on the static grid, mean over the 36 cells and fifteen seeds, for the four cached targets they can be trained on. Horizontal lines are the offline orders, which do not train through that model. Left: excess task risk; right: D_B.

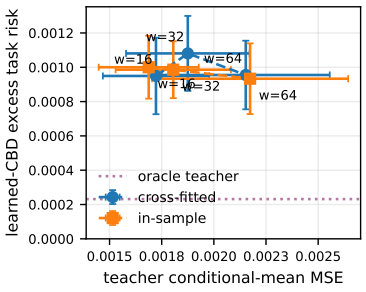
Feasible-output students against training-model error. Table 13 tabulates Figure 7. With the exact model the four feasible-output variants lie within a factor of 3.3 of one another, and with a wrong model within 15%, so the target they are trained on matters far less than the model they carry: at $\pm 25\%$ curvature error every one of them is 20 to 45 times worse than offline CBD, and at $\pm 50\%$, 60 to 250 times.

Table 13: Excess task risk of the feasible-output students (mean over 36 cells and fifteen seeds) against the relative curvature error of the differentiable model they train through, with the offline orders for reference.

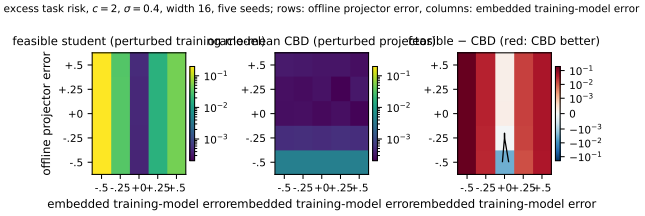
Student	-0.5	-0.25	0	$+0.25$	$+0.5$
Feasible (RAW targets)	7.03×10^{-2}	1.23×10^{-2}	1.85×10^{-4}	6.18×10^{-3}	1.90×10^{-2}
Feasible (PTF targets)	8.08×10^{-2}	1.39×10^{-2}	3.33×10^{-4}	6.72×10^{-3}	2.04×10^{-2}
Feasible (CBD targets)	8.08×10^{-2}	1.37×10^{-2}	1.87×10^{-4}	6.73×10^{-3}	2.06×10^{-2}
Feasible (oracle CBD targets)	7.94×10^{-2}	1.32×10^{-2}	1.02×10^{-4}	6.84×10^{-3}	2.08×10^{-2}
RAW (offline)			4.55×10^{-4}		
PTF (offline)			5.25×10^{-4}		
CBD (offline)			3.19×10^{-4}		
Oracle CBD (offline)			2.98×10^{-4}		

Pre-projection infeasibility. The distance between a student's raw output and its deployed output, $\mathbb{E}\|\widehat{f} - \Pi_{\mathcal{R}}(\widehat{f})\|^2$, is 2.1×10^{-2} for RAW at every width, 0.6–1.0×10^{-3} for PTF and 0.2–1.8×10^{-4} for CBD. RAW's student is therefore doing almost all of its work through the deployment projection, and its good task risk on this smooth set is a statement about the projection's benignity, not about the student's output being feasible.

C.3 Teacher sweep



(a) Teacher quality at $n = 512$, $c = 2$, $\sigma = 0.4$.



(b) Projector error against training-model error at $c = 2$, $\sigma = 0.4$.

Figure 8: **CBD's practical error terms.** Five seeds. Left: learned-CBD excess task risk against measured teacher conditional-mean error for three teacher widths and two teacher protocols, with the oracle-teacher level dotted. Right: excess task risk of the feasible-output student, whose differentiable training model is perturbed, and of oracle-mean CBD, whose offline projector is perturbed, over a 5×5 grid of parameter errors; both are deployed through the true plant. The third panel is their difference with the zero contour in black (red: CBD better).

Table 14 gives the sweep behind Figure 8: 512 training contexts, $c = 2$, $\sigma = 0.4$, a width-16 student trained for 600 epochs, and five-fold cross-fitted or in-sample teacher ensembles trained for 800 epochs with patience 100 on the training loss. At this sample size teacher width does not reduce conditional-mean error, and learned CBD's excess risk is flat within one standard error across widths while remaining four times the oracle-teacher level. Cross-fitting and in-sample fitting are indistinguishable here.

Sample size is the lever. Table 15 extends the sweep to training sizes 256 to $4{,}096$ and teacher widths 16 to 256, cross-fitted, five seeds. Reading down a column, more data reduces the teacher's error eight-fold and learned CBD's excess risk six-fold, from 1.8×10^{-3} to 2.8×10^{-4} against an oracle level of 2.1–2.3×10^{-4} that does not depend on n. Reading across a row, width hurts at every size up to $1{,}024$, where the width-256 teacher is the worst of the four by 30 to 65%, and helps only at $4{,}096$, where it is the best. The cost of a learned teacher, read as the oracle-minus-learned contrast, is therefore governed by sample size at these scales, and the recipe that closes it is more data, not a bigger network.

Table 14: Teacher sweep at $n = 512$: teacher conditional-mean MSE, learned-CBD excess task risk and D_B (mean $\pm$ std. over five seeds), and teacher training time.

Teacher	width	teacher MSE	CBD excess risk	CBD D_B	teacher (s)
cross-fitted	16	0.00172 ± 0.00057	0.00095 ± 0.00050	0.00105 ± 0.00047	2.6
cross-fitted	32	0.00188 ± 0.00066	0.00108 ± 0.00049	0.00120 ± 0.00047	2.8
cross-fitted	64	0.00215 ± 0.00090	0.00095 ± 0.00045	0.00108 ± 0.00044	3.5
in-sample	16	0.00169 ± 0.00054	0.00100 ± 0.00041	0.00111 ± 0.00038	2.9
in-sample	32	0.00181 ± 0.00062	0.00099 ± 0.00037	0.00110 ± 0.00043	3.0
in-sample	64	0.00217 ± 0.00105	0.00093 ± 0.00046	0.00105 ± 0.00045	3.8
oracle teacher	—	0	0.00023 ± 0.00003	0.00024 ± 0.00003	0

Table 15: Cross-fitted teacher sweep over training size and teacher width (five-seed means): teacher conditional-mean MSE, learned-CBD excess task risk, and the oracle-teacher level at the same size.

n	quantity	width 16	width 32	width 64	width 256	oracle teacher
256	teacher MSE	3.24×10^{-3}	3.35×10^{-3}	3.64×10^{-3}	4.71×10^{-3}	0
	CBD excess risk	1.77×10^{-3}	1.74×10^{-3}	1.90×10^{-3}	2.21×10^{-3}	2.28×10^{-4}
512	teacher MSE	1.72×10^{-3}	1.88×10^{-3}	2.15×10^{-3}	2.84×10^{-3}	0
	CBD excess risk	9.49×10^{-4}	1.08×10^{-3}	9.55×10^{-4}	1.07×10^{-3}	2.32×10^{-4}
1024	teacher MSE	7.96×10^{-4}	7.78×10^{-4}	8.31×10^{-4}	1.19×10^{-3}	0
	CBD excess risk	5.54×10^{-4}	6.36×10^{-4}	6.44×10^{-4}	6.79×10^{-4}	2.28×10^{-4}
4096	teacher MSE	3.98×10^{-4}	3.62×10^{-4}	3.47×10^{-4}	2.87×10^{-4}	0
	CBD excess risk	2.84×10^{-4}	3.20×10^{-4}	3.15×10^{-4}	3.18×10^{-4}	2.14×10^{-4}

C.4 Exact versus approximate feasible layers

The model-quality screen isolates two model errors, and its arms should be read by the information each receives. Its CBD arm is *oracle-mean* CBD: its labels are the oracle conditional mean passed through an offline projector whose curvature is perturbed, there is no learned teacher, and it is deployed through the exact true projector. Its feasible-output arm trains through a differentiable model whose curvature is perturbed and is evaluated by passing its predicted coordinate through the true plant. The screen therefore tests embedded training-model misspecification against offline projector misspecification, not general robustness to an inaccurate deployment model. The two errors vary independently over $\{-0.5, -0.25, 0, 0.25, 0.5\}$ at $c = 2$, $\sigma = 0.4$, 1,024 training contexts, a width-16 student and five seeds; Table 16 gives the full grid. Functional errors are recorded separately because equal parameter perturbations need not cause equal output changes. Three regularities are visible. The feasible-output student does not use the offline projector, so its entries vary down a column only because the screen draws each cell's initialization seed from both errors; its columns span three orders of magnitude. Oracle-mean CBD does not use the training model, so its entries vary along a row for the same reason, and the Oracle CBD block, which uses neither, shows the size of that seed-to-seed variation directly. Oracle-mean CBD's rows are asymmetric: overestimated curvature is harmless, underestimated curvature is not, because the flatter projector places labels on the wrong side of the true curve. The robust ensemble, which averages projections over five curvatures spread $\pm 30\%$ around the offline one, is worse than nominal CBD at every grid point, since the average of nearest points on different parabolas is not the nearest point on any of them. These numbers diagnose sensitivity, not a theorem that CBD tolerates every type of misspecification.

C.5 Trajectory benchmark

Settings. A context $x = (a, f, \phi)$ is the amplitude $a \in [0.55, 0.95]$, frequency $f \in [0.7, 2.0]$ cycles per horizon and phase $\phi \in [0, 2\pi)$ of a desired two-channel oscillation, drawn uniformly. With $\theta_t = 2\pi f\,(t-1)/(T-1) + \phi$ and $A = (1 + 0.45)\,a$, where 0.45 is the unreachable scale, the conditional mean is $\mu_t(x) = \big(A \sin\theta_t,\ 0.72A\cos(\theta_t + 0.25) + 0.18\sin 2\theta_t\big)$ for $t = 1, \ldots, T$, and each observed target adds independent Gaussian noise of standard deviation 0.08 to every coordinate. 512 training and 128 test contexts, $T = 24$ time steps; students of width 8, 16, 32 trained

Table 16: Excess task risk (five-seed mean) on the model-error screen. Rows: curvature error of the offline projector (oracle-mean CBD); columns: curvature error of the differentiable model the feasible-output student trains through. Both arms are deployed through the true plant.

offline \ online	-0.5	-0.25	0	$+0.25$	$+0.5$
*Feasible (*RAW *targets)*					
-0.5	1.96×10^{-1}	3.18×10^{-2}	4.21×10^{-4}	1.63×10^{-2}	4.75×10^{-2}
-0.25	1.97×10^{-1}	3.18×10^{-2}	4.44×10^{-4}	1.63×10^{-2}	4.75×10^{-2}
$+0$	1.97×10^{-1}	3.18×10^{-2}	4.51×10^{-4}	1.63×10^{-2}	4.75×10^{-2}
$+0.25$	1.96×10^{-1}	3.18×10^{-2}	4.58×10^{-4}	1.63×10^{-2}	4.75×10^{-2}
$+0.5$	1.96×10^{-1}	3.17×10^{-2}	4.76×10^{-4}	1.63×10^{-2}	4.75×10^{-2}
CBD					
-0.5	3.96×10^{-3}	4.02×10^{-3}	3.99×10^{-3}	4.01×10^{-3}	4.03×10^{-3}
-0.25	5.04×10^{-4}	5.15×10^{-4}	4.94×10^{-4}	5.30×10^{-4}	5.03×10^{-4}
$+0$	2.67×10^{-4}	2.74×10^{-4}	2.45×10^{-4}	2.73×10^{-4}	2.17×10^{-4}
$+0.25$	2.70×10^{-4}	2.60×10^{-4}	2.84×10^{-4}	2.05×10^{-4}	3.21×10^{-4}
$+0.5$	3.35×10^{-4}	3.18×10^{-4}	3.04×10^{-4}	3.03×10^{-4}	2.63×10^{-4}
Robust CBD					
-0.5	9.38×10^{-3}	9.36×10^{-3}	9.35×10^{-3}	9.42×10^{-3}	9.36×10^{-3}
-0.25	3.89×10^{-3}	3.86×10^{-3}	3.83×10^{-3}	3.87×10^{-3}	3.90×10^{-3}
$+0$	2.90×10^{-3}	3.03×10^{-3}	2.85×10^{-3}	2.90×10^{-3}	2.89×10^{-3}
$+0.25$	3.04×10^{-3}	3.08×10^{-3}	3.04×10^{-3}	3.02×10^{-3}	3.07×10^{-3}
$+0.5$	3.55×10^{-3}	3.49×10^{-3}	3.48×10^{-3}	3.54×10^{-3}	3.49×10^{-3}
Oracle CBD					
-0.5	2.26×10^{-4}	2.31×10^{-4}	2.52×10^{-4}	2.87×10^{-4}	2.44×10^{-4}
-0.25	2.28×10^{-4}	2.44×10^{-4}	2.21×10^{-4}	2.45×10^{-4}	2.16×10^{-4}
$+0$	2.08×10^{-4}	2.61×10^{-4}	2.35×10^{-4}	2.20×10^{-4}	2.83×10^{-4}
$+0.25$	2.67×10^{-4}	3.01×10^{-4}	2.27×10^{-4}	3.24×10^{-4}	2.51×10^{-4}
$+0.5$	2.26×10^{-4}	2.32×10^{-4}	2.54×10^{-4}	2.24×10^{-4}	2.22×10^{-4}

for 600 epochs (patience 90, lr 4×10^{-3}); a width-128 teacher fitted in-sample for 900 epochs, and a five-fold cross-fitted ensemble of the same class for the cross-fitted arm; the label projector of Appendix B.2 and the three-candidate reference built there; fifteen seeds. The plant maps a command sequence $u_{1:T} \in [-1,1]^{T\times 2}$ to an output trajectory as follows: the secondary command is clipped to $\pm(1-0.55\,|u_{t,1}|)$ (coupling limit); each command passes a dead zone of width 0.12, $d = \operatorname{sign}(u)\max(|u|-0.12,0)/0.88$; the state follows the asymmetric first-order response

$$s_t = s_{t-1} + a_t\,(d_t - s_{t-1}), \qquad a_t = 0.24 + (0.42-0.24)\,\operatorname{clip}\Big(\tfrac{|d_t|-|s_{t-1}|}{w} + \tfrac{1}{2},\,0,\,1\Big), \qquad w = 0.1, \tag{26}$$

with rise rate 0.42, fall rate 0.24 and $s_0 = 0$; and the second output coordinate is $s_{t,2}(1 + 0.2\,s_{t,1})$. The feasible-output students embed a copy of the plant whose dead zone is offset by $\{0, 0.05, 0.12\}$ and whose coupling limit is scaled by $\{-25\%, 0, +25\%\}$; the 3×3 grid with fifteen seeds, three widths and eleven arms, plus the two warm-started arms, the two long-budget cold arms and the four compute-matched arms below on the exact plant only, gives $4{,}815$ rows, of which the 855 at zero error are Table 25.

Continuity and existence of a nearest feasible trajectory. The theory assumes a closed reachable set and an attained nearest point. The plant of Eq. 26 satisfies this:

Proposition 5 (Compact reachable set)**.** *Let $\Phi : [-1,1]^{T\times 2} \to \mathbb{R}^{T\times 2}$ be the realization map of Eq. 26 with $w > 0$. Then Φ is continuous, the reachable set $\mathcal{R} = \Phi([-1,1]^{T\times 2})$ is compact, and for every $y \in \mathbb{R}^{T\times 2}$ the distance $\operatorname{dist}(y, \mathcal{R})$ is attained.*

Proof. Φ is a finite composition of continuous maps: the coupling clip is a minimum and maximum of continuous functions; the dead-zone map is continuous, because $\max(|u| - 0.12, 0)$ vanishes on a neighbourhood of $u = 0$ where the sign is discontinuous; the rate a_t is a clipped affine function of the continuous quantity $|d_t| - |s_{t-1}|$; the state update is affine in (d_t, s_{t-1}) with a continuous coefficient; and the output coupling is a polynomial. The domain is compact, so its continuous image is compact, hence closed and bounded, and a continuous function on a compact set attains its infimum. □

Uniqueness is not claimed, since $\mathcal{R}$ need not be convex, and the label projector remains an approximate minimizer (Appendix B.2). With the hard comparison $|d_t| \geq |s_{t-1}|$ in place of the ramp, that is Eq. 26 with $w = 0$, the switch an earlier draft of this paper used, the realization map is discontinuous at an opposite-sign switch: with a first primary command 0.56, so that $d_1 = 0.5$ and $s_1 = 0.21$, a second command whose drive is $d_2 = -0.21 + \eta$ gives $s_2 = 0.1092 + 0.24\eta$ as $\eta \downarrow 0$, whereas the drive -0.21 itself, or any drive of larger magnitude, takes the rise branch and gives $s_2 = 0.0336$; the trajectory $((0.21, 0), (0.1092, 0))$ lies in the closure of that reachable set and is not attained, so no nearest feasible trajectory exists for targets that approach it. That plant is retained only for the diagnostic comparison below.

The two-link arm of the second plant is also discontinuous, and its own inelastic stop is the reason. A joint whose proposed position overshoots a limit is clamped there and has its velocity zeroed, while a proposal that stays inside by any margin keeps its velocity, so the state jumps across a stop-crossing threshold and the coupled dynamics carry the jump into the recorded end-effector path. In the hard regime at the settings of Table 22, with the shoulder commanded to zero and the elbow held at a constant for the first seven samples and at -1 afterwards, the first upper-stop event occurs at $a_* = 0.9249834847765$, where one elbow velocity tends to 1.1544 and the other to zero; the realized path differs by 0.0163 for commands $a_* \pm 10^{-6}$ and by the same 0.0163 for $a_* \pm 10^{-10}$, so the one-sided limits do not meet and no choice of which branch owns exact equality removes the gap. Proposition 5 therefore does not extend to that plant: its reachable set is bounded, but we do not claim it closed or the distance to it attained. Where it is not, the constrained Bayes target is to be read on the closure $\overline{\mathcal{R}}$, which has the same distance function, and the projectors on that plant are numerical searches toward it, not certified projections. Nothing in Stage 2 rests on that claim. The label projector there is a finite-budget search, every reported contrast is between two such searches on the same targets, and the reference-free criterion needs no projection at all; the check of Table 24 on the first plant, where a discontinuous switch changed no conclusion, is the empirical form of the same point. A continuous stop model would restore the guarantee and is the change to make if the theoretical connection is wanted on this plant.

The reachable set is not band-limited. A static nonlinearity maps a band-limited command to an output carrying harmonics outside that band, so the reachable set in trajectory space cannot be summarised by a per-frequency mask and a bandwidth figure for the plant does not bound what it can realise. We therefore treat $\mathcal{R}$ pointwise in trajectory space. The plant's state couples the time steps of one trajectory, but every trajectory starts from the same rest state ($s_0 = 0$), so the reachable set is one fixed, history-coupled set of trajectories rather than a context-dependent $\mathcal{R}(x)$; the student sees the same context the projector conditions on, and no hidden state enters one and not the other.

Projector budget and residual. Table 17 validates the projector budget. On unreachable targets the tracking error plateaus by 120 steps (0.0652 against 0.0650 at 240), which is the budget every arm uses; initializing the commands at zero instead of at the target lands in a basin seven times worse whatever the budget, because the gradient with respect to a command inside the dead zone vanishes and a zero start never leaves it, so the initialization is part of the projector's definition. Restarts from the target initialization are identical, since that initialization is deterministic, and the table shows them so; only the mixed initialization (the target and independent random draws) probes different basins. On known-reachable targets, obtained by projecting the training targets once and re-projecting the realized trajectories from scratch, the residual is $(1.6 \pm 0.5) \times 10^{-4}$ per coordinate, 0.05% of the trajectory energy and 0.5% of the unreachable-target tracking error of the same problem. This is a near-idempotence check on trajectories the projector itself produced, not a certificate of global nearest-point accuracy on unreachable means or student predictions, which is why the reference $\tilde{h}$ of Appendix B.2 is built separately, with a larger budget and three candidates, and why Table 25 reports the reference-free $\mathbb{E}\|g - \mu\|^2$ as well.

Table 17: Projector validation on unreachable targets ($T = 16$, 64 targets, five seeds): tracking MSE against the Adam step budget for three initializations and one or three restarts, and wall-clock at 120 steps.

init.	restarts	15	30	60	120	240	time at 120 (s)
target	1	0.0726	0.0666	0.0651	0.0646	0.0644	1.28
target	3	0.0726	0.0666	0.0651	0.0646	0.0644	4.05
mixed	1	0.0726	0.0666	0.0651	0.0646	0.0644	1.00
mixed	3	0.0726	0.0665	0.0648	0.0643	0.0640	4.95
zero	1	0.4760	0.4760	0.4760	0.4760	0.4760	1.24
zero	3	0.4760	0.4760	0.4760	0.4760	0.4760	4.62

Full table and paired differences. Table 25 lists every arm at every width under both criteria with the reference-free $\mathbb{E}\|g - \mu\|^2$ and the per-seed cost, and Table 26 gives the paired seed differences at every width under both criteria, with a check on every comparison decided by the 95% paired t-test over fifteen seeds. Three rows are the ones the main text singles out: RAW against oracle CBD at width 8 (decided under task risk, absent under D_B), RAW against learned CBD at width 16 (decided under task risk, $|t| = 2.1$ under D_B), and the warm-started feasible student against RAW at width 32 (not decided under task risk, decided under D_B). The cold-started feasible-output rows show that embedding the exact plant costs 0.032 ± 0.004 against RAW and 0.015 ± 0.002 against offline CBD at width 32, in every seed; the warm-started rows show the same student ahead of both. The reference $\tilde{h}$ tracks μ to 0.0322 against the label projector's 0.0328, an improvement of 6.0×10^{-4}; in 74% of the test contexts it is the regularized long run of Appendix B.2, in 22% the unregularized continuation and in 4% the label projector's own output, and it differs from the label projector's output by 7.6×10^{-4} per coordinate in mean square. The residual of 1.6×10^{-4} measured on re-projected reachable trajectories is a near-idempotence check and not a bound on this difference.

Warm-started feasible-output students, and what the warm start is made of. The two warm arms are the same students, on raw and on CBD targets, trained by a hybrid recipe: their command network is first fitted, with the student's optimizer and epoch budget and no plant in the loop, to the commands the label projector found for that target (the PTF inverse of the raw targets, the inverse of the CBD targets), and then trained end to end through the plant with the usual budget. They therefore receive command supervision that a cold start does not, and twice its parameter updates; their rows

are charged the inverse they imitate (the raw-target arm pays the PTF inverse's time and cache, Table 4) and they run on the exact plant only. Table 18 splits the gain with two further arms and one intermediate score: a cold start with twice the end-to-end budget, and the frozen command-imitation network scored on the held-out contexts before any end-to-end training. Imitation alone accounts for most of the gain. The frozen network beats the cold student in every seed at every width, by 0.018 to 0.033 of excess task risk, is not distinguished from RAW at widths 8 and 16 on fifteen seeds (-0.0037 ± 0.0031 and $+0.0001 \pm 0.0008$), and is behind it at width 32 ($+0.0029 \pm 0.0004$, every seed). End-to-end training through the plant then improves on it further, by 0.023, 0.0068 and 0.0036 on raw targets at widths 8, 16 and 32 (every seed, $|t| \geq 13$), and that second stage is what puts the student ahead of RAW. Doubling the cold budget instead helps at width 8 (by 0.032 and 0.023 on raw and CBD targets, decided) and little at widths 16 and 32, and leaves the cold student behind the warm one by 0.017 to 0.025 at every width, decided, so the number of updates is not the explanation. We read the result as the optimization half of the "accurate and started in the right basin" condition of Section 7: the exact differentiable plant is the best parameterization on this benchmark once the search starts near a good solution, and the inverse-generated commands that PTF and CBD construct anyway are worth more as supervision for the command network than as targets for an output network.

Table 18: What the warm start is made of (trajectory plant, fifteen seeds, mean $\pm$ std., in units of 10^{-2}): the feasible-output student on raw and on CBD targets under four recipes, both criteria, and the per-seed cost at width 32. "Command imitation only" is the warm arm's command network after its imitation stage and before any end-to-end training, scored on the held-out contexts; its cost is the warm arm's without the end-to-end stage.

	width 8		width 16		width 32		cost
Recipe ($\times 10^{-2}$)	task risk	D_B	task risk	D_B	task risk	D_B	(s)
feasible-output student on RAW *targets*							
cold, 600 epochs	7.38 ± 1.49	5.33 ± 1.11	3.65 ± 0.79	2.74 ± 0.60	3.86 ± 1.37	2.90 ± 1.00	7.2
cold, 1,200 epochs	4.22 ± 1.15	3.14 ± 0.81	3.24 ± 0.94	2.44 ± 0.66	3.10 ± 1.08	2.35 ± 0.80	14.4
imitation only, 600 epochs	4.04 ± 0.83	3.10 ± 0.64	1.43 ± 0.17	1.14 ± 0.13	0.93 ± 0.07	0.75 ± 0.06	2.0
imitation + 600 epochs (warm)	$\mathbf{1.73 \pm 0.27}$	$\mathbf{1.34 \pm 0.22}$	$\mathbf{0.75 \pm 0.06}$	$\mathbf{0.54 \pm 0.04}$	$\mathbf{0.57 \pm 0.04}$	$\mathbf{0.39 \pm 0.03}$	9.2
feasible-output student on CBD *targets*							
cold, 600 epochs	6.22 ± 1.35	4.15 ± 0.91	3.33 ± 0.75	2.24 ± 0.52	2.71 ± 0.64	1.85 ± 0.47	14.1
cold, 1,200 epochs	3.95 ± 1.03	2.67 ± 0.75	2.70 ± 0.75	1.85 ± 0.57	2.30 ± 0.61	1.60 ± 0.45	21.4
imitation only, 600 epochs	3.78 ± 0.87	2.91 ± 0.63	1.44 ± 0.16	1.16 ± 0.13	0.92 ± 0.05	0.75 ± 0.05	7.4
imitation + 600 epochs (warm)	$\mathbf{1.85 \pm 0.38}$	$\mathbf{1.28 \pm 0.30}$	$\mathbf{0.83 \pm 0.06}$	$\mathbf{0.52 \pm 0.05}$	$\mathbf{0.63 \pm 0.04}$	$\mathbf{0.38 \pm 0.03}$	14.5

Table 19: Paired seed differences behind Table 18 (excess task risk; mean $\pm$ s.e., in parentheses the number of the fifteen seeds in which the first recipe is better, a check for a decided comparison).

Contrast (first $-$ second), excess task risk	width 8	width 16	width 32
RAW *targets*			
cold 1,200 $-$ cold 600	-0.0316 ± 0.0034 (15)✓	-0.0041 ± 0.0020 (11)	-0.0076 ± 0.0021 (13)✓
imitation only $-$ cold 600	-0.0334 ± 0.0039 (15)✓	-0.0222 ± 0.0019 (15)✓	-0.0293 ± 0.0034 (15)✓
imitation only $-$ RAW	-0.0037 ± 0.0031 (11)	$+0.0001 \pm 0.0008$ (6)	$+0.0029 \pm 0.0004$ (0)✓
warm $-$ imitation only	-0.0231 ± 0.0017 (15)✓	-0.0068 ± 0.0003 (15)✓	-0.0036 ± 0.0001 (15)✓
warm $-$ cold 1,200	-0.0248 ± 0.0030 (15)✓	-0.0249 ± 0.0023 (15)✓	-0.0253 ± 0.0027 (15)✓
CBD *targets*			
cold 1,200 $-$ cold 600	-0.0227 ± 0.0026 (15)✓	-0.0062 ± 0.0015 (13)✓	-0.0041 ± 0.0012 (12)✓
imitation only $-$ cold 600	-0.0244 ± 0.0026 (15)✓	-0.0189 ± 0.0017 (15)✓	-0.0179 ± 0.0016 (15)✓
imitation only $-$ CBD	-0.0290 ± 0.0046 (15)✓	-0.0067 ± 0.0010 (15)✓	-0.0027 ± 0.0006 (14)✓
warm $-$ imitation only	-0.0193 ± 0.0015 (15)✓	-0.0061 ± 0.0003 (15)✓	-0.0030 ± 0.0001 (15)✓
warm $-$ cold 1,200	-0.0210 ± 0.0024 (15)✓	-0.0188 ± 0.0018 (15)✓	-0.0168 ± 0.0015 (15)✓

Compute-matched alternatives to the warm start. The warm start could be a training schedule that any equally funded optimizer matches. We fixed two alternatives in the package before running them (`STAGE1_PREDECLARED.md` in the results directory, left as written and carrying a dated amendment for the same two wordings as Stage 2's Part D): a *multi-start with learning-rate tuning*, four cold restarts, each from its own independent initialization, two at learning rate 4×10^{-3} and two at 8×10^{-3}, each trained for 40 epochs, the two with the lower training loss continued for 40, the better of those to the end of the budget, Adam's state kept throughout; and a *plant curriculum*, the same cold student trained for 60 epochs on each of four easier plants (dead zone $\lambda \cdot 0.12$, coupling

limit $\lambda \cdot 0.55$, fall rate $0.42 + \lambda(0.24 - 0.42)$ for $\lambda = 0, \frac{1}{4}, \frac{1}{2}, \frac{3}{4}$; at $\lambda = 0$ a clipped symmetric first-order low-pass) and then on the true plant. Neither imitates inverse commands and neither uses held-out data for any choice. Each is given 721 plant-batch calls, which on raw targets is exactly what the warm start spends — the 121 scored iterates of the inverse whose commands it imitates plus its 600 end-to-end epochs — and costs 0.92 to 0.98 of its wall time. That ledger counts the calls the warm arm consumes. The released projector also runs one further forward pass to realize its best commands, which this arm discards because it takes the commands and not the realized output, so an instrumented count of everything executed reads 722 against the alternatives' 721, 0.14% more; the exact match is a match of scored iterates, and the conclusion drawn from these controls does not turn on the difference. On CBD targets the match is not exact, and in the alternatives' favour: the target is itself the output of an inverse, so every arm of that family pays its 122 calls (121 scored iterates and the realization that is the target) before training, the warm arm's imitated commands coming free with it, and the alternatives then spend 843 calls against the warm arm's 722, 16.8% more, at 1.06 to 1.08 of its wall time (Table 20, whose call column separates the two stages). Exact matching of the scored iterates is therefore claimed for the raw-target family, which is the predeclared primary one; the CBD-target contrasts are secondary and their alternatives are the better funded side. The two primary contrasts, warm minus multi-start and warm minus curriculum under excess task risk, pooled over the three widths within each seed and Holm-corrected, were declared in advance, with the rule that a failure of either would narrow the paper's reading of the warm start to a schedule result. Both are decided in the warm start's favour, by 0.0264 ± 0.0017 and 0.0255 ± 0.0018, in every seed and at every width (Table 21); the secondary CBD-target family, where the alternatives hold the larger budget, gives the same picture (0.0231 ± 0.0013 and 0.0236 ± 0.0019, every seed). The alternatives are not weak: each beats the cold 600-epoch student in every seed, pooled, and reaches the level of the cold 1,200-epoch student ($+0.0014 \pm 0.0012$ and $+0.0004 \pm 0.0012$ against it, not decided), so the budget spent on restarts, a larger step or an easier plant buys what the same budget spent on more epochs buys, and the inverse's commands buy the rest. That last comparison is pooled over the widths and holds for the raw-target family only: at width 8 both alternatives are decided *worse* than the 1,200-epoch cold student ($+0.0057 \pm 0.0019$ and $+0.0077 \pm 0.0034$), and both CBD-target alternatives are decided worse than their own 1,200-epoch control even pooled ($+0.0043 \pm 0.0011$ and $+0.0048 \pm 0.0015$). The main text therefore states only the pooled comparison against the 600-epoch cold student and the contrast with the hybrid, which is the predeclared one. Both alternatives remain behind RAW at widths 16 and 32 (every seed) and are not distinguished from it at width 8. The boundary of this evidence is the plant: the alternatives are two natural optimizers, not all of them, and a different dynamics family is the test of the next paragraph.

Table 20: Compute-matched alternatives to the warm start (trajectory plant, fifteen seeds, mean $\pm$ std., in units of 10^{-2}): the feasible-output student on raw and on CBD targets cold for 600 and 1,200 epochs, under the budget-matched multi-start and the plant curriculum, and warm-started; both criteria, the plant-batch calls each recipe spends and the per-seed cost at width 32, measured in one run. A call count written as a sum separates the inverse a recipe pays for before training from its end-to-end epochs: on raw targets only the warm start pays one (121 scored iterates, matching the alternatives' 721 exactly; the projector's final realization, which that arm discards, is the 0.14% discussed in the text), whereas on CBD targets every recipe pays the shared 122-call target inverse and the alternatives end up 16.8% above the warm start.

	width 8		width 16		width 32		plant	cost
Recipe ($\times 10^{-2}$)	task risk	D_B	task risk	D_B	task risk	D_B	calls	(s)
feasible-output student on RAW *targets*								
cold, 600 epochs	7.38 ± 1.49	5.33 ± 1.11	3.65 ± 0.79	2.74 ± 0.60	3.86 ± 1.37	2.90 ± 1.00	600	7.2
cold, 1,200 epochs	4.22 ± 1.15	3.14 ± 0.81	3.24 ± 0.94	2.44 ± 0.66	3.10 ± 1.08	2.35 ± 0.80	1200	14.4
multi-start, lr-tuned, matched budget	4.78 ± 1.17	3.57 ± 0.89	3.18 ± 0.55	2.43 ± 0.45	3.02 ± 0.71	2.30 ± 0.56	721	8.8
plant curriculum, matched budget	4.99 ± 1.31	3.69 ± 0.94	2.95 ± 0.80	2.29 ± 0.61	2.76 ± 0.62	2.18 ± 0.48	721	8.8
imitation + 600 epochs (warm)	$\mathbf{1.73 \pm 0.27}$	$\mathbf{1.34 \pm 0.22}$	$\mathbf{0.75 \pm 0.06}$	$\mathbf{0.54 \pm 0.04}$	$\mathbf{0.57 \pm 0.04}$	$\mathbf{0.39 \pm 0.03}$	121+600	9.2
feasible-output student on CBD *targets*								
cold, 600 epochs	6.22 ± 1.35	4.15 ± 0.91	3.33 ± 0.75	2.24 ± 0.52	2.71 ± 0.64	1.85 ± 0.47	122+600	14.1
cold, 1,200 epochs	3.95 ± 1.03	2.67 ± 0.75	2.70 ± 0.75	1.85 ± 0.57	2.30 ± 0.61	1.60 ± 0.45	122+1200	21.4
multi-start, lr-tuned, 721-call budget	4.97 ± 0.98	3.32 ± 0.70	2.74 ± 0.60	1.88 ± 0.44	2.53 ± 0.60	1.76 ± 0.44	122+721	15.7
plant curriculum, 721-call budget	5.65 ± 1.63	3.83 ± 1.20	2.65 ± 0.62	1.89 ± 0.47	2.10 ± 0.43	1.54 ± 0.35	122+721	15.4
imitation + 600 epochs (warm)	$\mathbf{1.85 \pm 0.38}$	$\mathbf{1.28 \pm 0.30}$	$\mathbf{0.83 \pm 0.06}$	$\mathbf{0.52 \pm 0.05}$	$\mathbf{0.63 \pm 0.04}$	$\mathbf{0.38 \pm 0.03}$	122+600	14.5

Table 21: Paired seed differences behind Table 20 (excess task risk; mean ± s.e., in parentheses the number of the fifteen seeds in which the first recipe is better; ✓ decided at 95%, ✓✓ the two predeclared primary contrasts, decided under the Holm correction). The primary family is the raw-target one; the CBD block repeats the same two contrasts as secondary tests, marked like every other secondary comparison. The pooled column averages the three widths within each seed before pairing.

Contrast (first − second), excess task risk	width 8	width 16	width 32	pooled over widths
RAW *targets*				
warm − multi-start	-0.0305 ± 0.0030 (15)✓	-0.0243 ± 0.0014 (15)✓	-0.0245 ± 0.0018 (15)✓	-0.0264 ± 0.0017 (15)✓✓
warm − curriculum	-0.0325 ± 0.0031 (15)✓	-0.0220 ± 0.0020 (15)✓	-0.0219 ± 0.0016 (15)✓	-0.0255 ± 0.0018 (15)✓✓
multi-start − cold 600	-0.0260 ± 0.0038 (15)✓	-0.0047 ± 0.0018 (11)✓	-0.0084 ± 0.0025 (13)✓	-0.0130 ± 0.0016 (15)✓
curriculum − cold 600	-0.0239 ± 0.0046 (13)✓	-0.0070 ± 0.0016 (12)✓	-0.0110 ± 0.0021 (14)✓	-0.0140 ± 0.0016 (15)✓
multi-start − cold 1,200	$+0.0056 \pm 0.0019$ (4)✓	-0.0007 ± 0.0019 (6)	-0.0008 ± 0.0018 (7)	$+0.0014 \pm 0.0012$ (5)
curriculum − cold 1,200	$+0.0077 \pm 0.0034$ (4)✓	-0.0029 ± 0.0015 (10)	-0.0034 ± 0.0020 (10)	$+0.0004 \pm 0.0012$ (8)
curriculum − multi-start	$+0.0021 \pm 0.0039$ (8)	-0.0023 ± 0.0014 (10)	-0.0026 ± 0.0013 (11)	-0.0010 ± 0.0015 (8)
multi-start − RAW	$+0.0037 \pm 0.0033$ (5)	$+0.0176 \pm 0.0012$ (0)✓	$+0.0238 \pm 0.0018$ (0)✓	$+0.0150 \pm 0.0015$ (0)✓
curriculum − RAW	$+0.0058 \pm 0.0037$ (6)	$+0.0153 \pm 0.0017$ (0)✓	$+0.0211 \pm 0.0016$ (0)✓	$+0.0141 \pm 0.0018$ (0)✓
CBD *targets (secondary)*				
warm − multi-start	-0.0312 ± 0.0024 (15)✓	-0.0191 ± 0.0014 (15)✓	-0.0190 ± 0.0015 (15)✓	-0.0231 ± 0.0013 (15)✓
warm − curriculum	-0.0380 ± 0.0040 (15)✓	-0.0183 ± 0.0015 (15)✓	-0.0147 ± 0.0011 (15)✓	-0.0236 ± 0.0019 (15)✓
multi-start − cold 600	-0.0124 ± 0.0033 (12)✓	-0.0059 ± 0.0011 (14)✓	-0.0018 ± 0.0010 (11)	-0.0067 ± 0.0010 (14)✓
curriculum − cold 600	-0.0057 ± 0.0044 (9)	-0.0068 ± 0.0017 (13)✓	-0.0062 ± 0.0009 (15)✓	-0.0062 ± 0.0016 (13)✓
multi-start − cold 1,200	$+0.0102 \pm 0.0031$ (2)✓	$+0.0003 \pm 0.0014$ (6)	$+0.0023 \pm 0.0014$ (5)	$+0.0043 \pm 0.0011$ (2)✓
curriculum − cold 1,200	$+0.0170 \pm 0.0037$ (2)✓	-0.0005 ± 0.0018 (7)	-0.0021 ± 0.0011 (10)	$+0.0048 \pm 0.0015$ (3)✓
curriculum − multi-start	$+0.0068 \pm 0.0048$ (4)	-0.0009 ± 0.0012 (10)	-0.0043 ± 0.0008 (13)✓	$+0.0005 \pm 0.0017$ (7)
multi-start − CBD	-0.0171 ± 0.0048 (13)✓	$+0.0063 \pm 0.0013$ (2)✓	$+0.0133 \pm 0.0013$ (0)✓	$+0.0008 \pm 0.0016$ (6)
curriculum − CBD	-0.0103 ± 0.0065 (10)	$+0.0054 \pm 0.0016$ (1)✓	$+0.0090 \pm 0.0010$ (0)✓	$+0.0014 \pm 0.0024$ (5)

A second plant: the two-link arm. The boundary just named is tested on a plant of a different kind (`arm_plant.py`): a contact-free planar two-link arm driven by joint torques and observed at its end effector, with rigid-body dynamics (a configuration-dependent mass matrix and Coriolis terms, no gravity), viscous joint damping, a torque limit, inelastic joint stops (a joint that reaches a stop stays there and its velocity is zeroed) and the coupled trigonometric forward kinematics of two uniform rods of length 0.5 and mass 0.5; semi-implicit Euler with three steps per observed sample, 24 samples over 1.5 s, the arm at rest in the home configuration $(0.6, 1.4)$ rad at the start. Commands are normalized torques in $[-1, 1]$ times the torque limit. Nothing in it is a dead zone; the flat regions of the realization map are the torque limit and the stops. The map is differentiable away from those boundaries but, unlike the first plant's, not continuous: the stop is inelastic, so a proposal that overshoots a limit is clamped and loses its velocity while one that stays inside by any margin keeps it, and the coupled dynamics turn that jump into a jump of the recorded path (the counterexample and its consequences are in the paragraph on continuity above). The demonstrations come from a *stronger* arm of the same kinematics (torque limit 6, damping 0.6, elbow range 0.15–2.9 rad) that follows a smooth joint-space path in closed loop (PD, gains 60 and 6, torque clamped to its limit); the path is the first plant's three-parameter family (amplitude 0.55–0.95 times 0.9 rad, frequency 0.7–2.0 cycles per horizon, uniform phase) about the home configuration, the conditional mean is the generator arm's end-effector path including its transient from rest, and the targets add isotropic Gaussian noise of standard deviation 0.04. The deployment arm is driven in open loop by the student's torques, so the demonstrations are unreachable for it wherever the path needs more torque, less damping or more elbow range than it has. Two regimes of the deployment arm were fixed on plant-level criteria alone, before any student was trained at them: *easy*, torque limit 4.5, damping 0.8, the generator's elbow range, where the label projector tracks the conditional mean to within the observation noise (0.71 times the noise variance over the thirty seeds; 23% of the generator's torque samples exceed the limit); and *hard*, torque limit 2, damping 1.6, elbow range 0.8–1.9 rad, where it tracks it about eight times worse than the noise variance (7.9 times; 49% of the samples exceed the limit, and on the check seeds a tenth of the desired elbow path leaves the range). The label projector and the reference are those of Appendix B.2, except that the inverse starts from zero torque, where every command direction has a live gradient. Thirteen arms run in both regimes at widths 8 and 32 with the first plant's recipes and settings: the three output students, the cold feasible-output student on raw and on CBD targets for 600 and for $1{,}200$ epochs, the two warm-started students, and the multi-start and the plant curriculum of Table 20 at the same 721-call budget (matching the warm arm's scored iterates exactly on raw targets and exceeding it by 16.8% on CBD targets, for the reason given there), the curriculum's stages now interpolating the deployment arm's torque limit, damping and joint ranges from the generator's values. The design was written into the package before the pilot (`STAGE2_PREDECLARED.md`, whose Part A we left as written and amended only by dated appendices: a Part C, the correction of its continuity sentence above, and a Part D, its restart wording

and the call convention of the next paragraph): the plant, the regimes and their criteria, the arms, the two primary contrasts, warm minus multi-start and warm minus curriculum under excess task risk averaged over the four regime-width conditions within each seed and Holm-corrected, and the sample-size rule. A pilot on five separate seeds supplied the variance and the smallest useful effect, the gain that doubling the cold budget buys (0.0027 averaged over the conditions); the rule, power 0.8 at that effect with the pilot standard deviation inflated to its upper 80% bound, asked for more than the cap of thirty seeds, so the confirmatory run uses thirty (power 0.79 at that effect) and no pilot row enters a table.

Both primary contrasts are decided in the warm start's favour, by 0.0041 ± 0.0002 against the multi-start and 0.0111 ± 0.0008 against the curriculum, in all thirty seeds and in each of the four conditions (Table 23); the secondary CBD-target family, where the alternatives hold 16.8% more plant calls, gives 0.0032 ± 0.0001 and 0.0164 ± 0.0007, again every seed. D_B and the reference-free error give the same decisions. The regimes did not separate the effect: the warm start beats the cold 600-epoch student in every seed in the easy regime as in the hard one (0.0066 ± 0.0007 and 0.0057 ± 0.0003, pooled over the widths), so the regime where the demonstrations are reachable to within the noise is not one where cold training suffices on this plant; the effect is smallest at width 32 in the easy regime (0.0032 ± 0.0002 over the cold student, 0.0027 ± 0.0002 over the multi-start) and largest at width 8. The multi-start is again a strong control, though not quite the equal of a doubled cold budget here: it stays 0.0006 ± 0.0001 behind the cold $1{,}200$-epoch student (decided, ahead in 6 of the thirty seeds) and beats the output student RAW in every seed; the curriculum, whose easy stages are here the generator arm's own dynamics, is worse than the cold 600-epoch student in both regimes and 0.0077 ± 0.0007 behind the cold $1{,}200$-epoch student pooled, so on this plant an easy-to-true schedule is not a good use of the budget. The arm also changes the picture below the warm start: the cold feasible-output student beats RAW in three of the four conditions (pooled 0.0035 ± 0.0006, 25 of 30 seeds) and loses to it only in the hard regime at width 32, so the exact feasible parameterization is competitive from a random start on a plant whose flat regions are a torque limit and joint stops rather than a dead zone, and the warm start adds to it. The three output students rank as on the first plant: RAW beats PTF in every condition ($+0.0052 \pm 0.0004$ pooled, PTF minus RAW) and CBD is not distinguished from PTF in any (Table 23). Costs, measured in the same run: the multi-start and the curriculum cost the warm start's wall time within 3% on raw targets (10–16% more on CBD targets); per seed the warm student costs 6.2 to 6.4 times RAW, the cold student 5.1 to 5.2, the $1{,}200$-epoch cold student 10.3 to 10.9, and the CBD-target warm student 2.7 to 2.9 times learned CBD; at deployment the feasible student's single realization takes about 20 ms per query batch against 8 to 9 s for an output arm's inverse (Table 22). The multi-start selected the doubled learning rate in 85 of 120 races in the easy regime and 116 of 120 in the hard one. The observed contrasts are fifteen and twenty standard errors from zero, well beyond what the sample-size rule was built for.

Table 22: The two-link arm plant (thirty seeds, mean $\pm$ std., in units of 10^{-2}): the output student and the feasible-output student on raw and on CBD targets under the recipes of Table 20, in the easy and the hard regime at widths 8 and 32, both criteria, the plant-batch calls each recipe spends (a sum separates the inverse paid for before training from the end-to-end epochs, as in Table 20) and the per-seed cost in the hard regime at width 32, measured in one run.

	easy, width 8		easy, width 32		hard, width 8		hard, width 32		plant	cost
Recipe ($\times 10^{-2}$)	task risk	D_B	task risk	D_B	task risk	D_B	task risk	D_B	calls	(s, hard w32)
RAW *targets*										
RAW output student (inverse at deployment)	2.25 ± 0.39	2.12 ± 0.37	0.57 ± 0.18	0.52 ± 0.17	1.46 ± 0.28	1.08 ± 0.20	0.27 ± 0.11	0.21 ± 0.08	–	9.2
feasible, cold, 600 epochs	1.29 ± 0.75	1.16 ± 0.70	0.48 ± 0.09	0.41 ± 0.08	0.76 ± 0.14	0.43 ± 0.10	0.61 ± 0.16	0.35 ± 0.10	600	47.9
feasible, cold, 1,200 epochs	0.58 ± 0.19	0.51 ± 0.17	0.31 ± 0.07	0.27 ± 0.06	0.62 ± 0.19	0.36 ± 0.12	0.56 ± 0.15	0.33 ± 0.09	1200	95.8
feasible, multi-start, lr-tuned, matched budget	0.63 ± 0.15	0.55 ± 0.14	0.43 ± 0.09	0.38 ± 0.09	0.64 ± 0.18	0.37 ± 0.11	0.60 ± 0.18	0.36 ± 0.12	721	58.0
feasible, plant curriculum, matched budget	1.64 ± 0.75	1.48 ± 0.71	0.59 ± 0.15	0.51 ± 0.14	2.09 ± 0.90	1.26 ± 0.53	0.82 ± 0.23	0.48 ± 0.16	721	58.3
feasible, imitation + 600 epochs (warm)	$\mathbf{0.30 \pm 0.04}$	$\mathbf{0.26 \pm 0.04}$	$\mathbf{0.16 \pm 0.02}$	$\mathbf{0.13 \pm 0.01}$	$\mathbf{0.15 \pm 0.02}$	$\mathbf{0.09 \pm 0.02}$	$\mathbf{0.07 \pm 0.01}$	$\mathbf{0.05 \pm 0.01}$	121+600	57.8
PTF *targets*										
PTF output student (inverse at deployment)	2.46 ± 0.32	2.27 ± 0.31	0.71 ± 0.22	0.61 ± 0.22	2.55 ± 0.23	1.44 ± 0.13	0.90 ± 0.22	0.44 ± 0.15	–	18.6
CBD *targets*										
CBD output student (inverse at deployment)	2.39 ± 0.40	2.20 ± 0.38	0.69 ± 0.12	0.59 ± 0.11	2.60 ± 0.26	1.46 ± 0.14	0.94 ± 0.18	0.48 ± 0.11	–	22.5
feasible, cold, 600 epochs	1.22 ± 0.45	1.05 ± 0.42	0.57 ± 0.13	0.46 ± 0.11	0.67 ± 0.13	0.24 ± 0.06	0.39 ± 0.07	0.15 ± 0.04	122+600	60.9
feasible, cold, 1,200 epochs	0.62 ± 0.14	0.51 ± 0.12	0.39 ± 0.06	0.30 ± 0.05	0.45 ± 0.05	0.16 ± 0.03	0.29 ± 0.05	0.12 ± 0.03	122+1200	109.3
feasible, multi-start, lr-tuned, 721-call budget	0.76 ± 0.16	0.63 ± 0.14	0.55 ± 0.09	0.44 ± 0.08	0.51 ± 0.10	0.19 ± 0.06	0.36 ± 0.06	0.14 ± 0.04	122+721	71.4
feasible, plant curriculum, 721-call budget	2.00 ± 0.82	1.79 ± 0.78	0.66 ± 0.09	0.53 ± 0.08	3.20 ± 0.61	1.71 ± 0.36	1.63 ± 0.47	0.75 ± 0.28	122+721	70.2
feasible, imitation + 600 epochs (warm)	$\mathbf{0.34 \pm 0.04}$	$\mathbf{0.28 \pm 0.04}$	$\mathbf{0.20 \pm 0.01}$	$\mathbf{0.16 \pm 0.01}$	$\mathbf{0.24 \pm 0.02}$	$\mathbf{0.09 \pm 0.01}$	$\mathbf{0.13 \pm 0.01}$	$\mathbf{0.05 \pm 0.01}$	122+600	61.9

The hard-switch plant. Table 24 reruns every arm on the hard-switch plant (Eq. 26 with $w = 0$) with the same seeds, widths, budgets, reference construction and deployment inverse. The best arm is the same at every width under both criteria, and the set of decided pairwise comparisons differs by

Table 23: Paired seed differences behind Table 22 (excess task risk; mean ± s.e., in parentheses the number of the thirty seeds in which the first recipe is better; ✓ decided at 95%, ✓✓ the two predeclared primary contrasts, decided under the Holm correction). The primary family is the raw-target one; the CBD block repeats the same two contrasts as secondary tests, with 16.8% more plant calls on the alternatives' side, and is marked like every other secondary comparison. The pooled column averages the four regime-width conditions within each seed before pairing.

Contrast (first − second), excess task risk	easy, w8	easy, w32	hard, w8	hard, w32	pooled over the four
RAW *targets*					
warm − multi-start	-0.0033 ± 0.0003 (30)✓	-0.0027 ± 0.0002 (30)✓	-0.0049 ± 0.0003 (30)✓	-0.0053 ± 0.0003 (30)✓	-0.0041 ± 0.0002 (30)✓✓
warm − curriculum	-0.0134 ± 0.0014 (30)✓	-0.0042 ± 0.0003 (30)✓	-0.0194 ± 0.0016 (30)✓	-0.0075 ± 0.0004 (30)✓	-0.0111 ± 0.0008 (30)✓✓
warm − cold 600	-0.0099 ± 0.0014 (30)✓	-0.0032 ± 0.0002 (30)✓	-0.0061 ± 0.0003 (30)✓	-0.0053 ± 0.0003 (30)✓	-0.0061 ± 0.0004 (30)✓
warm − cold 1,200	-0.0028 ± 0.0003 (30)✓	-0.0015 ± 0.0001 (30)✓	-0.0047 ± 0.0004 (30)✓	-0.0049 ± 0.0003 (30)✓	-0.0035 ± 0.0002 (30)✓
warm − RAW	-0.0195 ± 0.0007 (30)✓	-0.0041 ± 0.0003 (30)✓	-0.0131 ± 0.0005 (30)✓	-0.0020 ± 0.0002 (30)✓	-0.0097 ± 0.0003 (30)✓
multi-start − cold 600	-0.0066 ± 0.0014 (26)✓	-0.0004 ± 0.0001 (25)✓	-0.0013 ± 0.0003 (27)✓	-0.0000 ± 0.0002 (14)	-0.0021 ± 0.0004 (29)✓
curriculum − cold 600	$+0.0035 \pm 0.0019$ (12)	$+0.0011 \pm 0.0002$ (5)✓	$+0.0133 \pm 0.0016$ (2)✓	$+0.0021 \pm 0.0005$ (7)✓	$+0.0050 \pm 0.0008$ (5)✓
multi-start − cold 1,200	$+0.0005 \pm 0.0004$ (10)	$+0.0012 \pm 0.0002$ (2)✓	$+0.0002 \pm 0.0003$ (14)	$+0.0004 \pm 0.0002$ (10)✓	$+0.0006 \pm 0.0001$ (6)✓
curriculum − cold 1,200	$+0.0106 \pm 0.0014$ (2)✓	$+0.0028 \pm 0.0003$ (0)✓	$+0.0147 \pm 0.0016$ (0)✓	$+0.0026 \pm 0.0005$ (5)✓	$+0.0077 \pm 0.0007$ (0)✓
curriculum − multi-start	$+0.0101 \pm 0.0014$ (0)✓	$+0.0015 \pm 0.0002$ (2)✓	$+0.0145 \pm 0.0016$ (0)✓	$+0.0022 \pm 0.0004$ (5)✓	$+0.0071 \pm 0.0007$ (0)✓
cold 600 − RAW	-0.0096 ± 0.0018 (25)✓	-0.0009 ± 0.0003 (21)✓	-0.0070 ± 0.0006 (30)✓	$+0.0033 \pm 0.0004$ (2)✓	-0.0035 ± 0.0006 (25)✓
CBD *targets (secondary)*					
warm − multi-start	-0.0042 ± 0.0003 (30)✓	-0.0035 ± 0.0001 (30)✓	-0.0027 ± 0.0002 (30)✓	-0.0023 ± 0.0001 (30)✓	-0.0032 ± 0.0001 (30)✓
warm − curriculum	-0.0166 ± 0.0015 (30)✓	-0.0045 ± 0.0002 (30)✓	-0.0296 ± 0.0011 (30)✓	-0.0150 ± 0.0009 (30)✓	-0.0164 ± 0.0007 (30)✓
warm − cold 600	-0.0088 ± 0.0008 (30)✓	-0.0037 ± 0.0002 (30)✓	-0.0043 ± 0.0002 (30)✓	-0.0026 ± 0.0001 (30)✓	-0.0048 ± 0.0003 (30)✓
warm − cold 1,200	-0.0028 ± 0.0002 (30)✓	-0.0018 ± 0.0001 (30)✓	-0.0022 ± 0.0001 (30)✓	-0.0016 ± 0.0001 (30)✓	-0.0021 ± 0.0001 (30)✓
warm − CBD	-0.0205 ± 0.0007 (30)✓	-0.0049 ± 0.0002 (30)✓	-0.0236 ± 0.0005 (30)✓	-0.0081 ± 0.0003 (30)✓	-0.0143 ± 0.0003 (30)✓
multi-start − cold 600	-0.0046 ± 0.0008 (28)✓	-0.0002 ± 0.0002 (15)	-0.0016 ± 0.0003 (26)✓	-0.0003 ± 0.0001 (22)✓	-0.0017 ± 0.0003 (29)✓
curriculum − cold 600	$+0.0079 \pm 0.0018$ (4)✓	$+0.0008 \pm 0.0002$ (6)✓	$+0.0253 \pm 0.0012$ (0)✓	$+0.0124 \pm 0.0008$ (0)✓	$+0.0116 \pm 0.0007$ (0)✓
multi-start − cold 1,200	$+0.0014 \pm 0.0003$ (6)✓	$+0.0016 \pm 0.0001$ (0)✓	$+0.0006 \pm 0.0002$ (11)✓	$+0.0007 \pm 0.0001$ (3)✓	$+0.0011 \pm 0.0001$ (2)✓
curriculum − cold 1,200	$+0.0138 \pm 0.0015$ (0)✓	$+0.0027 \pm 0.0002$ (0)✓	$+0.0274 \pm 0.0011$ (0)✓	$+0.0133 \pm 0.0009$ (0)✓	$+0.0143 \pm 0.0007$ (0)✓
curriculum − multi-start	$+0.0125 \pm 0.0015$ (1)✓	$+0.0011 \pm 0.0002$ (5)✓	$+0.0269 \pm 0.0012$ (0)✓	$+0.0127 \pm 0.0009$ (0)✓	$+0.0133 \pm 0.0007$ (0)✓
cold 600 − CBD	-0.0117 ± 0.0009 (28)✓	-0.0012 ± 0.0003 (23)✓	-0.0193 ± 0.0004 (30)✓	-0.0055 ± 0.0004 (30)✓	-0.0094 ± 0.0003 (30)✓
output students (inverse at deployment)					
PTF − RAW	$+0.0021 \pm 0.0008$ (11)✓	$+0.0013 \pm 0.0005$ (11)✓	$+0.0109 \pm 0.0006$ (0)✓	$+0.0062 \pm 0.0004$ (0)✓	$+0.0052 \pm 0.0004$ (0)✓
CBD − PTF	-0.0007 ± 0.0008 (17)	-0.0002 ± 0.0004 (13)	$+0.0004 \pm 0.0006$ (12)	$+0.0005 \pm 0.0006$ (12)	$+0.0000 \pm 0.0004$ (14)
CBD − RAW	$+0.0014 \pm 0.0010$ (15)	$+0.0012 \pm 0.0004$ (9)✓	$+0.0114 \pm 0.0006$ (0)✓	$+0.0067 \pm 0.0004$ (0)✓	$+0.0052 \pm 0.0004$ (0)✓

one pair per width (oracle against learned CBD at width 8 is decided only on the continuous plant; robust CBD against CBD at width 16 and the warm-started student against RAW at width 32 only on the hard-switch plant). The per-arm means differ by at most 0.0084, for the cold-started student on raw targets at width 8, and by at most 0.0006 for any output-emitting arm. The discontinuity therefore did not create the results of Section 5.2; what it did affect is the scaling family below, which is not even consistent with a continuous map on that plant.

Table 24: Trajectory plant with the continuous rate switch of Eq. 26 against the hard switch ($w = 0$), excess task risk, fifteen-seed means on identical seeds; a check next to a mean marks a comparison with RAW decided on that plant, and the last column marks a decided difference between the plants.

Method	continuous plant	hard-switch plant	continuous − hard	decided
width 8				
RAW	0.0441	0.0444	-0.0003 ± 0.0001	✓
PTF	0.0584✓	0.0588✓	-0.0005 ± 0.0004	
CBD	0.0668✓	0.0652✓	$+0.0016 \pm 0.0018$	
Oracle CBD	0.0541✓	0.0548✓	-0.0007 ± 0.0003	
Feasible (RAW targets)	0.0738✓	0.0653✓	$+0.0084 \pm 0.0032$	✓
Feasible (CBD targets)	0.0622	0.0575	$+0.0047 \pm 0.0028$	
Feasible, warm start (RAW targets)	**0.0173**✓	**0.0171**✓	$+0.0002 \pm 0.0002$	
Feasible, warm start (CBD targets)	0.0185✓	0.0172✓	$+0.0013 \pm 0.0007$	
width 16				
RAW	0.0142	0.0143	-0.0002 ± 0.0001	✓
PTF	0.0251✓	0.0252✓	-0.0001 ± 0.0001	
CBD	0.0211✓	0.0213✓	-0.0003 ± 0.0001	✓
Oracle CBD	0.0222✓	0.0223✓	-0.0002 ± 0.0002	
Feasible (RAW targets)	0.0365✓	0.0335✓	$+0.0030 \pm 0.0022$	
Feasible (CBD targets)	0.0333	0.0286	$+0.0047 \pm 0.0009$	✓
Feasible, warm start (RAW targets)	**0.0075**✓	**0.0074**✓	$+0.0001 \pm 0.0001$	
Feasible, warm start (CBD targets)	0.0083✓	0.0085✓	-0.0002 ± 0.0001	
width 32				
RAW	0.0064	0.0066	-0.0002 ± 0.0001	✓
PTF	0.0133✓	0.0133✓	-0.0001 ± 0.0001	
CBD	0.0119✓	0.0125✓	-0.0006 ± 0.0002	✓
Oracle CBD	0.0121✓	0.0124✓	-0.0003 ± 0.0001	✓
Feasible (RAW targets)	0.0386✓	0.0321✓	$+0.0065 \pm 0.0017$	✓
Feasible (CBD targets)	0.0271	0.0253	$+0.0018 \pm 0.0012$	
Feasible, warm start (RAW targets)	**0.0057**	**0.0058**✓	-0.0001 ± 0.0001	
Feasible, warm start (CBD targets)	0.0063	0.0065	-0.0002 ± 0.0001	✓

Table 25: Trajectory plant, all arms, fifteen seeds, mean ± std. of excess task risk and D_B against the reference $\tilde{h}$ of Appendix B.2, the reference-free $\mathbb{E}\|g-\mu\|^2$, and the per-seed wall-clock cost (labels, teacher where one is used, student and deployment projection, seconds; for a feasible-output student the deployment term is its single timed realization, and Table 1 uses the same itemization). Bold is the best mean per width and column. Feasible-output rows embed the exact plant; cold rows start from a random initialization, warm rows from a command network fitted to the label projector's inverse commands, and the 2×-budget rows are cold starts with 1,200 end-to-end epochs.

Method	excess task risk	D_B	$\mathbb{E}\|g-\mu\|^2$	cost (s)
width 8				
RAW	0.0441 ± 0.0089	0.0346 ± 0.0066	0.0763 ± 0.0087	1.7
PTF	0.0584 ± 0.0144	0.0382 ± 0.0094	0.0906 ± 0.0149	3.1
CBD	0.0668 ± 0.0196	0.0447 ± 0.0138	0.0990 ± 0.0195	8.5
CBD (cross-fitted)	0.0628 ± 0.0144	0.0414 ± 0.0104	0.0950 ± 0.0142	19.8
Oracle CBD	0.0541 ± 0.0107	0.0357 ± 0.0072	0.0863 ± 0.0108	3.2
Robust CBD	0.0634 ± 0.0182	0.0414 ± 0.0122	0.0956 ± 0.0188	14.4
Robust oracle CBD	0.0617 ± 0.0136	0.0404 ± 0.0095	0.0939 ± 0.0137	8.6
Feasible (RAW targets)	0.0738 ± 0.0149	0.0533 ± 0.0111	0.1060 ± 0.0155	7.3
Feasible (PTF targets)	0.0737 ± 0.0202	0.0488 ± 0.0142	0.1059 ± 0.0197	8.4
Feasible (CBD targets)	0.0622 ± 0.0135	0.0415 ± 0.0091	0.0944 ± 0.0137	13.9
Feasible (oracle CBD targets)	0.0622 ± 0.0120	0.0413 ± 0.0086	0.0944 ± 0.0120	8.7
Feasible, warm start (RAW targets)	$\mathbf{0.0173 \pm 0.0027}$	0.0134 ± 0.0022	$\mathbf{0.0495 \pm 0.0028}$	9.2
Feasible, warm start (CBD targets)	0.0185 ± 0.0038	$\mathbf{0.0128 \pm 0.0030}$	0.0507 ± 0.0047	14.5
Feasible, cold, 2× budget (RAW targets)	0.0422 ± 0.0115	0.0314 ± 0.0081	0.0744 ± 0.0113	14.2
Feasible, cold, 2× budget (CBD targets)	0.0395 ± 0.0103	0.0267 ± 0.0075	0.0717 ± 0.0101	21.3
Feasible, multi-start, matched budget (RAW targets)	0.0478 ± 0.0117	0.0357 ± 0.0089	0.0800 ± 0.0114	8.5
Feasible, multi-start, matched budget (CBD targets)	0.0497 ± 0.0098	0.0332 ± 0.0070	0.0819 ± 0.0100	15.4
Feasible, plant curriculum, matched budget (RAW targets)	0.0499 ± 0.0131	0.0369 ± 0.0094	0.0821 ± 0.0120	8.7
Feasible, plant curriculum, matched budget (CBD targets)	0.0565 ± 0.0163	0.0383 ± 0.0120	0.0887 ± 0.0162	15.4
width 16				
RAW	0.0142 ± 0.0029	0.0124 ± 0.0021	0.0464 ± 0.0039	1.7
PTF	0.0251 ± 0.0037	0.0162 ± 0.0024	0.0573 ± 0.0048	3.1
CBD	0.0211 ± 0.0039	0.0137 ± 0.0026	0.0533 ± 0.0048	8.5
CBD (cross-fitted)	0.0216 ± 0.0037	0.0140 ± 0.0024	0.0538 ± 0.0048	19.6
Oracle CBD	0.0222 ± 0.0036	0.0143 ± 0.0023	0.0544 ± 0.0042	3.1
Robust CBD	0.0243 ± 0.0048	0.0155 ± 0.0031	0.0565 ± 0.0056	14.2
Robust oracle CBD	0.0245 ± 0.0038	0.0155 ± 0.0025	0.0567 ± 0.0047	8.5
Feasible (RAW targets)	0.0365 ± 0.0079	0.0274 ± 0.0060	0.0687 ± 0.0074	7.2
Feasible (PTF targets)	0.0332 ± 0.0079	0.0219 ± 0.0057	0.0654 ± 0.0080	8.9
Feasible (CBD targets)	0.0333 ± 0.0075	0.0224 ± 0.0052	0.0655 ± 0.0074	14.0
Feasible (oracle CBD targets)	0.0344 ± 0.0072	0.0229 ± 0.0052	0.0666 ± 0.0069	8.6
Feasible, warm start (RAW targets)	$\mathbf{0.0075 \pm 0.0006}$	0.0054 ± 0.0004	$\mathbf{0.0397 \pm 0.0022}$	8.9
Feasible, warm start (CBD targets)	0.0083 ± 0.0006	$\mathbf{0.0052 \pm 0.0005}$	0.0405 ± 0.0023	14.5
Feasible, cold, 2× budget (RAW targets)	0.0324 ± 0.0094	0.0244 ± 0.0066	0.0646 ± 0.0090	14.4
Feasible, cold, 2× budget (CBD targets)	0.0270 ± 0.0075	0.0185 ± 0.0057	0.0592 ± 0.0081	20.8
Feasible, multi-start, matched budget (RAW targets)	0.0318 ± 0.0055	0.0243 ± 0.0045	0.0640 ± 0.0052	8.7
Feasible, multi-start, matched budget (CBD targets)	0.0274 ± 0.0060	0.0188 ± 0.0044	0.0596 ± 0.0058	15.2
Feasible, plant curriculum, matched budget (RAW targets)	0.0295 ± 0.0080	0.0229 ± 0.0061	0.0617 ± 0.0081	8.5
Feasible, plant curriculum, matched budget (CBD targets)	0.0265 ± 0.0062	0.0189 ± 0.0047	0.0587 ± 0.0061	15.3
width 32				
RAW	0.0064 ± 0.0014	0.0062 ± 0.0011	0.0386 ± 0.0022	1.8
PTF	0.0133 ± 0.0033	0.0085 ± 0.0023	0.0455 ± 0.0040	3.2
CBD	0.0119 ± 0.0024	0.0078 ± 0.0016	0.0441 ± 0.0032	8.6
CBD (cross-fitted)	0.0114 ± 0.0018	0.0074 ± 0.0011	0.0435 ± 0.0032	19.7
Oracle CBD	0.0121 ± 0.0014	0.0077 ± 0.0011	0.0443 ± 0.0022	3.2
Robust CBD	0.0123 ± 0.0011	0.0079 ± 0.0007	0.0445 ± 0.0025	14.3
Robust oracle CBD	0.0133 ± 0.0027	0.0085 ± 0.0016	0.0455 ± 0.0041	8.6
Feasible (RAW targets)	0.0386 ± 0.0137	0.0290 ± 0.0100	0.0708 ± 0.0135	7.2
Feasible (PTF targets)	0.0263 ± 0.0063	0.0180 ± 0.0047	0.0585 ± 0.0062	8.5
Feasible (CBD targets)	0.0271 ± 0.0064	0.0185 ± 0.0047	0.0593 ± 0.0064	14.1
Feasible (oracle CBD targets)	0.0281 ± 0.0094	0.0191 ± 0.0066	0.0603 ± 0.0096	8.9
Feasible, warm start (RAW targets)	$\mathbf{0.0057 \pm 0.0004}$	0.0039 ± 0.0003	$\mathbf{0.0379 \pm 0.0023}$	9.2
Feasible, warm start (CBD targets)	0.0063 ± 0.0004	$\mathbf{0.0038 \pm 0.0003}$	0.0385 ± 0.0023	14.5
Feasible, cold, 2× budget (RAW targets)	0.0310 ± 0.0108	0.0235 ± 0.0080	0.0632 ± 0.0104	14.4
Feasible, cold, 2× budget (CBD targets)	0.0230 ± 0.0061	0.0160 ± 0.0045	0.0552 ± 0.0057	21.4
Feasible, multi-start, matched budget (RAW targets)	0.0302 ± 0.0071	0.0230 ± 0.0056	0.0624 ± 0.0069	8.8
Feasible, multi-start, matched budget (CBD targets)	0.0253 ± 0.0060	0.0176 ± 0.0044	0.0575 ± 0.0061	15.7
Feasible, plant curriculum, matched budget (RAW targets)	0.0276 ± 0.0062	0.0218 ± 0.0048	0.0597 ± 0.0063	8.8
Feasible, plant curriculum, matched budget (CBD targets)	0.0210 ± 0.0043	0.0154 ± 0.0035	0.0532 ± 0.0043	15.4

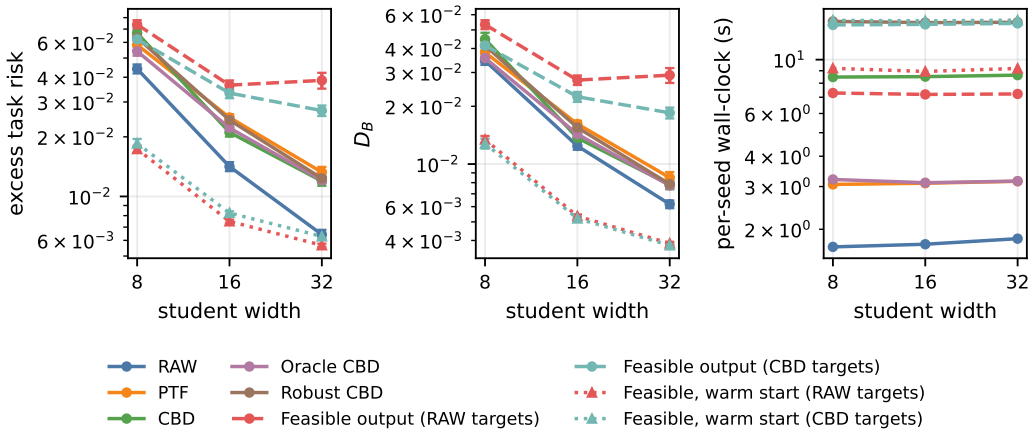


Figure 9: **Trajectory plant, every arm.** Excess task risk (left) and D_B (centre) against student width for the three orders, oracle CBD, robust CBD and the feasible-output students with the exact plant embedded, trained from a random initialization (dashed) or by the hybrid recipe (dotted); right, the per-seed wall-clock of labels, teacher where one is used, student and deployment projection, from the same rows as Table 25. Error bars are standard errors over fifteen seeds.

Table 26: Trajectory plant: paired seed differences (first minus second; mean ± s.e. and, in parentheses, the number of the fifteen seeds in which the first is better; a check marks a comparison decided at the two-sided 95% level) at each width under excess task risk and under D_B.

Comparison	criterion	width 8	width 16	width 32
Oracle CBD − RAW	task risk	$+0.0100 \pm 0.0030$ (1)✓	$+0.0080 \pm 0.0012$ (1)✓	$+0.0056 \pm 0.0005$ (0)✓
	D_B	$+0.0011 \pm 0.0022$ (9)	$+0.0019 \pm 0.0008$ (4)✓	$+0.0016 \pm 0.0004$ (2)✓
CBD − RAW	task risk	$+0.0227 \pm 0.0055$ (2)✓	$+0.0069 \pm 0.0009$ (0)✓	$+0.0055 \pm 0.0007$ (1)✓
	D_B	$+0.0102 \pm 0.0040$ (3)✓	$+0.0013 \pm 0.0006$ (5)	$+0.0016 \pm 0.0005$ (2)✓
PTF − RAW	task risk	$+0.0143 \pm 0.0041$ (2)✓	$+0.0109 \pm 0.0008$ (0)✓	$+0.0068 \pm 0.0009$ (0)✓
	D_B	$+0.0036 \pm 0.0029$ (6)	$+0.0038 \pm 0.0006$ (1)✓	$+0.0023 \pm 0.0006$ (1)✓
CBD − PTF	task risk	$+0.0085 \pm 0.0043$ (4)	-0.0040 ± 0.0014 (13)✓	-0.0013 ± 0.0009 (8)
	D_B	$+0.0066 \pm 0.0030$ (4)✓	-0.0025 ± 0.0010 (13)✓	-0.0007 ± 0.0006 (8)
Oracle CBD − CBD	task risk	-0.0127 ± 0.0058 (11)✓	$+0.0011 \pm 0.0012$ (8)	$+0.0001 \pm 0.0005$ (7)
	D_B	-0.0091 ± 0.0041 (11)✓	$+0.0006 \pm 0.0009$ (6)	-0.0001 ± 0.0003 (7)
CBD (cross-fitted) − CBD	task risk	-0.0040 ± 0.0060 (9)	$+0.0005 \pm 0.0007$ (7)	-0.0006 ± 0.0007 (8)
	D_B	-0.0033 ± 0.0043 (9)	$+0.0003 \pm 0.0005$ (5)	-0.0004 ± 0.0005 (8)
Robust CBD − CBD	task risk	-0.0035 ± 0.0060 (7)	$+0.0032 \pm 0.0015$ (6)	$+0.0004 \pm 0.0006$ (6)
	D_B	-0.0033 ± 0.0043 (7)	$+0.0018 \pm 0.0010$ (6)	$+0.0001 \pm 0.0004$ (6)
Feasible (RAW) − RAW	task risk	$+0.0297 \pm 0.0048$ (1)✓	$+0.0223 \pm 0.0020$ (0)✓	$+0.0321 \pm 0.0035$ (0)✓
	D_B	$+0.0187 \pm 0.0036$ (1)✓	$+0.0150 \pm 0.0015$ (0)✓	$+0.0229 \pm 0.0026$ (0)✓
Feasible (CBD) − CBD	task risk	-0.0046 ± 0.0039 (8)	$+0.0122 \pm 0.0016$ (0)✓	$+0.0152 \pm 0.0015$ (0)✓
	D_B	-0.0033 ± 0.0028 (8)	$+0.0087 \pm 0.0011$ (0)✓	$+0.0107 \pm 0.0011$ (0)✓
Warm feasible (RAW) − Feasible (RAW)	task risk	-0.0565 ± 0.0039 (15)✓	-0.0290 ± 0.0020 (15)✓	-0.0329 ± 0.0035 (15)✓
	D_B	-0.0399 ± 0.0029 (15)✓	-0.0220 ± 0.0015 (15)✓	-0.0252 ± 0.0025 (15)✓
Warm feasible (CBD) − Feasible (CBD)	task risk	-0.0436 ± 0.0029 (15)✓	-0.0250 ± 0.0018 (15)✓	-0.0209 ± 0.0016 (15)✓
	D_B	-0.0287 ± 0.0019 (15)✓	-0.0172 ± 0.0013 (15)✓	-0.0147 ± 0.0012 (15)✓
Warm feasible (RAW) − RAW	task risk	-0.0268 ± 0.0024 (15)✓	-0.0067 ± 0.0007 (15)✓	-0.0008 ± 0.0004 (10)
	D_B	-0.0211 ± 0.0018 (15)✓	-0.0071 ± 0.0006 (15)✓	-0.0023 ± 0.0003 (15)✓
Warm feasible (CBD) − CBD	task risk	-0.0483 ± 0.0047 (15)✓	-0.0128 ± 0.0010 (15)✓	-0.0057 ± 0.0006 (15)✓
	D_B	-0.0319 ± 0.0033 (15)✓	-0.0085 ± 0.0007 (15)✓	-0.0040 ± 0.0004 (15)✓
Cold 2× (RAW) − Feasible (RAW)	task risk	-0.0316 ± 0.0034 (15)✓	-0.0041 ± 0.0020 (11)	-0.0076 ± 0.0021 (13)✓
	D_B	-0.0219 ± 0.0024 (15)✓	-0.0030 ± 0.0015 (10)	-0.0056 ± 0.0014 (13)✓
Cold 2× (CBD) − Feasible (CBD)	task risk	-0.0227 ± 0.0026 (15)✓	-0.0062 ± 0.0015 (13)✓	-0.0041 ± 0.0012 (12)✓
	D_B	-0.0147 ± 0.0017 (15)✓	-0.0039 ± 0.0010 (13)✓	-0.0025 ± 0.0009 (11)✓
Warm feasible (RAW) − Cold 2× (RAW)	task risk	-0.0248 ± 0.0030 (15)✓	-0.0249 ± 0.0023 (15)✓	-0.0253 ± 0.0027 (15)✓
	D_B	-0.0180 ± 0.0021 (15)✓	-0.0191 ± 0.0016 (15)✓	-0.0196 ± 0.0020 (15)✓
Warm feasible (CBD) − Cold 2× (CBD)	task risk	-0.0210 ± 0.0024 (15)✓	-0.0188 ± 0.0018 (15)✓	-0.0168 ± 0.0015 (15)✓
	D_B	-0.0140 ± 0.0017 (15)✓	-0.0133 ± 0.0014 (15)✓	-0.0122 ± 0.0011 (15)✓

Plant misspecification inside the feasible layer. Figure 10 shows the feasible-output students' excess risk relative to offline CBD at the same width over the dead-zone and coupling error grid, for the cold-started students. Two things are visible. Even the exact plant (bottom-centre cell) loses to offline CBD at widths 16 and 32, by factors 1.6–1.7 and 2.3–3.2, so on this plant the EMBED order's deficit from a random start is an optimization deficit, not a modelling one, which the warm-started rows of Table 25 and the decomposition of Table 18 confirm; and misspecification compounds it, up to $4.2\times$ at width 32 with the dead zone overestimated by 0.12 and the coupling by 25%. Overestimating the dead zone is worse than misestimating the coupling, and the student trained on CBD targets is less sensitive than the one trained on raw targets in 25 of the 27 cells (equal in one and reversed by 0.05 in one, both at width 8). At width 8, where every offline arm is poor, the cold-started feasible students are within 0.93–1.41 of offline CBD across the grid but lose to RAW, which is the best standard-budget cold arm at that width too (Table 26).

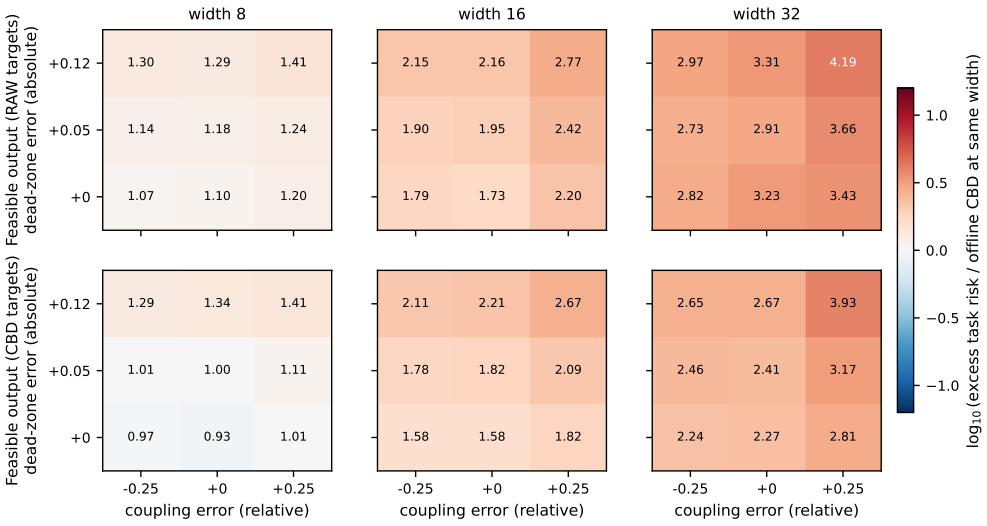


Figure 10: **Feasible-output students under plant misspecification.** Excess task risk relative to offline CBD at the same width (cell labels are the ratio; colour is its logarithm) for the student trained on raw targets (top) and on CBD targets (bottom), over dead-zone offset and relative coupling error. Fifteen seeds per cell.

What the deployment inverse does to an error. RAW's output student beats oracle CBD's at every width (Table 1), although oracle CBD has neither teacher error nor projection-order bias. The four output students were run again on the exact plant with each prediction recorded before the deployment inverse, every other column reproducing Table 25 bit for bit (Table 27). Each arm is read against one *diagnostic anchor* — the conditional mean μ for RAW, and the label projector's output $\widehat{\Pi}(\mu)$ for the projected orders — and the post-inverse error is the distance between the images of those same two points under the operator that is deployed, so the ratio of the two is that operator's measured contraction of the arm's discrepancy from its anchor rather than a comparison across references. The anchors are diagnostic reference functions, not every arm's training target: μ and $\widehat{\Pi}(\mu)$ are the population targets of RAW and of oracle CBD, but PTF trains toward $\mathbb{E}[\widehat{\Pi}(Y) \mid X]$ and learned CBD toward $\widehat{\Pi}(T_\psi(X))$, so their pre-inverse error also contains projection-order bias or teacher error. The on-set anchor is a fixed point of the operator to within 1.3×10^{-4}, two percent of the smallest post-inverse entry.

Two observations follow; neither measures how hard each regression problem is. First, as fractions of their anchor's variance the four pre-inverse errors lie within a third of one another at every width (0.125–0.155, 0.052–0.060 and 0.025–0.034) and do not order the arms the same way twice: oracle CBD's student is closest at width 8, learned CBD's at 16 and RAW's at 32. Second, the operator treats the arms differently. It displaces RAW's predictions by a mean squared distance of 2.0 to 2.8×10^{-2}, against 2.2 to 2.9×10^{-3} for the projected orders — 7.0, 10.0 and 12.5 times as far in mean square at the three widths — and the conditional mean itself by 3.3×10^{-2}. Because the

inverse returns a reachable output, each displacement is an upper bound on the squared distance to the reachable set, not that distance. RAW's error contracts to 50–54% and the projected orders' to 80–93%; RAW's ratio is the lowest of the four at every width, in all nine paired comparisons, in every one of the fifteen seeds and decided in each.

This is a contraction, not a decomposition. A nearest-point map discards the component of a displacement normal to the set *and* rescales the tangential one: at distance d outside a sphere of radius ρ the tangential factor is $\rho/(\rho + d)$, so an error with no normal part at all is still shortened, and inside the reach on the concave side the factor exceeds one (Appendix A.4). The measured ratios are the net of both effects and do not say what fraction of each error was normal; separating them would need directionally controlled perturbations or the operator's Jacobian, neither of which we measured. The local formula of Appendix A.3 is a different quantity: it expands a projection *at* a point of the set, where the Jacobian is exactly the tangential projector. Finally, $\widehat{\Pi}$ here is the regularized finite-budget inverse of Appendix B.2, so these are properties of the operator actually deployed and not of exact Euclidean projection.

Table 27: Output students on the exact trajectory plant (fifteen seeds, mean $\pm$ std., $\times 10^{-2}$ except the two ratios). Each student is compared with a diagnostic anchor, μ for RAW and $\widehat{\Pi}(\mu)$ for the projected orders, which is the population training target of RAW and of oracle CBD only (text). *pre-inverse* is the mean squared error to the anchor, *post-inverse* the mean squared distance between the images of student and anchor under the deployment inverse, and *post/pre* their ratio averaged over seeds, the operator's net contraction and not a normal–tangential split; *inverse displacement* is $\mathbb{E}\|f - \widehat{\Pi}(f)\|^2$, an upper bound on the student's squared distance to the reachable set; the last column is the pre-inverse error as a fraction of the anchor's variance.

	MSE to the diagnostic anchor				
Output student ($\times 10^{-2}$)	pre-inverse	post-inverse	post/pre	inverse displacement	pre / anchor var.
width 8					
RAW	6.66 ± 1.10	3.47 ± 0.67	0.52	2.04 ± 0.21	0.143
PTF	4.13 ± 0.99	3.82 ± 0.95	0.92	0.27 ± 0.04	0.133
CBD	4.84 ± 1.41	4.49 ± 1.40	0.92	0.28 ± 0.05	0.155
Oracle CBD	3.91 ± 0.78	3.57 ± 0.73	0.91	0.29 ± 0.03	0.125
width 16					
RAW	2.50 ± 0.41	1.25 ± 0.22	0.50	2.50 ± 0.15	0.053
PTF	1.87 ± 0.28	1.62 ± 0.24	0.87	0.24 ± 0.03	0.060
CBD	1.62 ± 0.30	1.37 ± 0.26	0.85	0.25 ± 0.03	0.052
Oracle CBD	1.68 ± 0.26	1.44 ± 0.24	0.85	0.25 ± 0.03	0.054
width 32					
RAW	1.17 ± 0.26	0.62 ± 0.11	0.54	2.82 ± 0.17	0.025
PTF	1.04 ± 0.26	0.85 ± 0.23	0.82	0.22 ± 0.03	0.034
CBD	0.98 ± 0.20	0.78 ± 0.16	0.80	0.22 ± 0.03	0.031
Oracle CBD	0.97 ± 0.12	0.78 ± 0.11	0.80	0.23 ± 0.02	0.031

The warm start with a misspecified embedded plant. The grid above trains only cold feasible students; the warm start of the main text had been run with the exact plant alone. With its design written into the package before the run (`WARM_MISSPEC_PREDECLARED.md`), the cold and warm students on raw and on CBD targets were run in the eight misspecified cells, fifteen seeds and three widths. Labels and the imitated commands come from the true plant, only the layer the student trains through is wrong, and every arm is deployed on the true plant (Table 28); the cold rows reproduce the grid's bit for bit. Averaged within seed over the eight cells and three widths, the warm student beats the cold one by 0.0438 ± 0.0021 on raw targets, the primary contrast, and by 0.0379 ± 0.0023 on CBD targets, in every seed, and it is ahead in all 24 (cell, width) combinations of both families, decided in every one. Against the RAW output student, whose deployment inverse uses the true plant and so does not see the misspecification, the raw-target warm student stays ahead at widths 8 and 16 (0.0251 ± 0.0024 and 0.0052 ± 0.0007, averaged over the eight cells, every seed) and not at width 32 ($+0.0007 \pm 0.0004$, not decided), where the CBD-target warm student falls behind RAW ($+0.0015 \pm 0.0003$, decided; last rows of Table 28): a wrong embedded plant costs the warm start its lead over RAW where that lead was smallest, and none of its lead over the cold start.

Commands or outputs: which product of the inverse supervises better. The warm start uses one of the two things an inverse solve returns. Holding that solve, its epoch budget and the second

Table 28: The warm start with a misspecified plant inside the feasible layer (fifteen seeds): excess task risk ($\times 10^{-2}$, mean over the three widths) of the cold and warm feasible students in each misspecified cell, and the paired warm-minus-cold difference (mean $\pm$ s.e., seeds in which warm is better, ✓ decided). The last three rows compare, per width, the warm student averaged over the eight cells with the RAW output student, whose deployment inverse uses the true plant (seeds in which warm is better). Labels and imitated commands use the true plant; all arms are deployed on it. Design fixed before the run.

error in the embedded plant		RAW targets			CBD targets		
dead zone	coupling	cold	warm	warm − cold	cold	warm	warm − cold
0	−0.25	4.76	1.13	-3.63 ± 0.21 (15)✓	4.15	1.23	-2.93 ± 0.17 (15)✓
0	+0.25	5.59	1.18	-4.41 ± 0.25 (15)✓	4.65	1.36	-3.29 ± 0.20 (15)✓
0.05	−0.25	4.97	1.12	-3.85 ± 0.20 (15)✓	4.49	1.17	-3.32 ± 0.20 (15)✓
0.05	0	5.15	1.03	-4.12 ± 0.26 (15)✓	4.46	1.11	-3.35 ± 0.19 (15)✓
0.05	+0.25	5.91	1.27	-4.64 ± 0.23 (15)✓	5.20	1.46	-3.73 ± 0.25 (15)✓
0.12	−0.25	5.59	1.13	-4.46 ± 0.22 (15)✓	5.42	1.14	-4.28 ± 0.28 (15)✓
0.12	0	5.70	1.09	-4.61 ± 0.20 (15)✓	5.61	1.17	-4.43 ± 0.31 (15)✓
0.12	+0.25	6.76	1.41	-5.35 ± 0.19 (15)✓	6.58	1.60	-4.98 ± 0.32 (15)✓
all eight		5.56	1.17	-4.38 ± 0.21 (15)✓	5.07	1.28	-3.79 ± 0.23 (15)✓
warm, averaged over the eight cells, minus the RAW *output student*							
width 8				-2.51 ± 0.24 (15)✓			-2.35 ± 0.22 (15)✓
width 16				-0.52 ± 0.07 (15)✓			-0.42 ± 0.07 (15)✓
width 32				$+0.07 \pm 0.04$ (3)			$+0.15 \pm 0.03$ (2)✓

stage fixed, we replaced the first stage with one that imitates the *outputs* the same commands realize: the whole student — command network and embedded plant — is fitted to $\widehat{\Pi}(y_i)$ on raw targets and to $\widehat{\Pi}(T_\psi(x_i))$ on CBD targets, with the plant in the loop, for the same number of epochs, after which the identical end-to-end stage runs. The design was written into the package before the run (`LABEL_REPRESENTATION_PREDECLARED.md`); the two cold arms in the same shards reproduce Table 25's rows bit for bit, all 90 of them. Budgets are not matched and the mismatch favours the new arm: its first stage calls the plant once per epoch, so it spends 1,200 plant-batch calls against the command stage's 600, and twice the training seconds (12.5–12.8 against 6.7–6.9 at every width).

Command imitation is the better first stage at every width, on both target families, in every one of the fifteen seeds (Table 29): pooled within seed over the three widths it is ahead by 0.0149 ± 0.0018 on raw targets, the primary contrast, and by 0.0181 ± 0.0019 on CBD targets, and the per-width contrasts are decided at every width under excess task risk and under D_B (0.0109 ± 0.0013 and 0.0128 ± 0.0013 pooled). The gap is already there before the shared second stage: the frozen first-stage networks, deployed on the true plant, differ by 0.0190 ± 0.0019 and 0.0238 ± 0.0020 pooled in the same direction. Output imitation is not useless — it beats the cold start by 0.0245 ± 0.0014 and 0.0117 ± 0.0014 pooled, so a first stage built from the inverse helps in either representation — but its loop contains the same plant as the end-to-end stage, so it inherits whatever obstacle the cold start faces, while the command stage regresses on the inverse's answer directly and never touches the plant. Which feature of the plant's gradients is responsible we did not isolate; the compute-matched curriculum of Table 20, which starts from a plant without dead zone or coupling limit, does not reproduce the warm start either. On this plant, then, the two products of one inverse solve are not interchangeable as supervision, and the cheaper one is the better one.

The size of the cross term, measured. The two criteria differ by $2\mathbb{E}\langle g - \tilde{h}, \tilde{h} - \mu\rangle$ (Appendix C.1), and how large that term is relative to the reference distance $D_{\tilde{h}} = \mathbb{E}\|g - \tilde{h}\|^2$ cannot be read off two significance decisions. We measure it along a controlled feasible family: from the reference commands $u^\star$ of $\tilde{h}$ on the test contexts, $g_\tau = \Phi(\mathrm{clip}(u^\star \pm \tau w))$ for unit random command directions w, four directions per context, five seeds, $\tau \in \{0.0025, \ldots, 0.08\}$. For each τ we record $D_{\tilde{h}}(\tau)$ and the cross term, split into its even part over $\pm w$, the perturbation response that survives averaging over direction, and its odd part, which is nonzero when the two one-sided responses differ or a descent direction remains. Neither part has a guaranteed sign: at an exact nearest point the sum of distance and cross term is nonnegative for every feasible g, which bounds the even part below by $-\frac{1}{2}\big(D(g_+) + D(g_-)\big)$ and not by zero, and on the parabola the even part is in fact negative, the

Table 29: One inverse solve, two representations of its answer as the first stage (fifteen seeds; excess task risk $\times 10^{-2}$, mean $\pm$ std., and the paired command-minus-output difference, mean $\pm$ s.e., seeds in which commands are better, ✓ decided). Same inverse, same first-stage epoch budget and identical second stage; the output stage additionally runs the plant once per epoch. Design fixed before the run.

Excess task risk ($\times 10^{-2}$)	cold	commands	outputs	commands − outputs
RAW *targets*				
width 8	7.38 ± 1.49	1.73 ± 0.27	2.89 ± 0.71	-1.16 ± 0.19 (15)✓
width 16	3.65 ± 0.79	0.75 ± 0.06	2.31 ± 0.82	-1.57 ± 0.21 (15)✓
width 32	3.86 ± 1.37	0.57 ± 0.04	2.32 ± 0.92	-1.75 ± 0.23 (15)✓
pooled over widths				-1.49 ± 0.18 (15)✓
CBD *targets*				
width 8	6.22 ± 1.35	1.85 ± 0.38	3.88 ± 1.15	-2.03 ± 0.26 (15)✓
width 16	3.33 ± 0.75	0.83 ± 0.06	2.44 ± 0.61	-1.62 ± 0.15 (15)✓
width 32	2.71 ± 0.64	0.63 ± 0.04	2.42 ± 0.80	-1.79 ± 0.20 (15)✓
pooled over widths				-1.81 ± 0.19 (15)✓

constant $-\kappa\|h^\star - \mu\|$ fraction of D that Appendix C.1 predicts (-0.06 to -0.09; Table 30). On the trajectory plant it is positive at every τ and large: 6.9 times $D_{\tilde{h}}$ at the smallest step and 2.6 times at the largest, so that the excess task risk of a small feasible perturbation of $\tilde{h}$ is dominated by the cross term and the two criteria measure different things there. Its scale dependence is not a single power. $D_{\tilde{h}}$ scales as $\tau^{2.0}$ throughout, while the slope of the even cross term between adjacent perturbation sizes rises from 1.41 at the largest steps to 1.93 at the smallest, approaching two as the step shrinks, which is what a mixture of smooth and boundary-crossing responses whose active sets change with τ would produce (40% of the reference commands are saturated and 10% lie in the dead zone). The odd part is 70% of the even part in absolute value at the smallest step and 3% at the largest. These are finite-range measurements on an approximate reference and clipped command-space perturbations: they establish that the cross term is substantial over the tested range and that its effective order varies with scale, not an asymptotic order. On the hard-switch plant the same family is not even consistent with a continuous map, $D_{\tilde{h}}$ scaling as $\tau^{0.8}$ and the even cross term changing sign with τ, because a perturbation of any size flips the rate branch in a fraction of contexts proportional to τ.

Table 30: The cross term along a controlled feasible family $g_\tau \to \tilde{h}$ (trajectory plant, five seeds, four random command directions per context and their negatives) and $g_\tau \to h^\star$ (parabola, exact projector, $\sigma = 0.4$): fitted log-log slopes in τ of the reference distance $D_{\tilde{h}}$ and of the even (symmetric) part of the cross term, over all τ and over $\tau \le 0.02$; the slope of the even part between each pair of adjacent perturbation sizes, from the largest step to the smallest; the ratio of the even part to $D_{\tilde{h}}$ at the largest and at the smallest step; the ratio of the odd to the even part at the smallest step; and the sign of the even part.

	fitted slope (all / ≤ 0.02)		even slope, adjacent τ	even/$D_{\tilde{h}}$	|odd|/|even|	sign of
Family	$D_{\tilde{h}}$	even cross	(largest step → smallest)	τ=0.08 / 0.0025	τ=0.0025	even
trajectory, continuous	1.99 / 1.99	1.70 / 1.84	1.41/1.58/1.72/1.85/1.93	+2.57 / +6.94	0.70	+ all τ
trajectory, hard switch	0.75 / 0.84	0.66 / 0.28	2.15/1.07/-0.44/0.84/0.25	+0.16 / −0.15	1.76	changes
parabola, $c = 0.5$	2.00 / 2.00	2.00 / 2.00	2.00/2.00/2.00/2.00/2.00	−0.08 / −0.08	0.00	− all τ
parabola, $c = 1$	2.00 / 2.00	2.00 / 2.00	2.00/2.00/2.00/2.00/2.00	−0.09 / −0.09	0.00	− all τ
parabola, $c = 2$	2.00 / 2.00	2.00 / 2.00	2.00/2.00/2.00/2.00/2.00	−0.06 / −0.06	0.00	− all τ

Cached labels move iterative plant optimization outside student training. The reported compute fields include target precomputation, teacher training, student training, deployment projection time, online plant calls, and target cache size (Table 4). Because cached targets can be reused across widths, seeds, and hyperparameters, both one-shot and amortized comparisons should be reported.

C.6 A PHYSICAL REACHABLE SET

The experiments above construct the geometry; this one measures it. We use motor-babble recordings from three silicone robot faces, two builds of one design and one of another, whose mouths are tracked as 40 landmarks and so span the 80-dimensional output space of Figure 11a. A recording drives the unit's actuated mouth degrees of freedom along a random walk, holds each command until the face has come to rest, and photographs the pose, so the cloud samples static reachable outputs

and not motion. A pose counts as at rest when the mean absolute pixel difference of the mouth region between consecutive camera frames stays below 2.5 (8-bit scale) twice in a row, after a minimum wait (80 ms on unit 1, 200 ms on units 2 and 3) and with a timeout (0.7 s and 1.0 s); unit 1 came to rest in 187 ms on average. The clouds hold 4,000, 2,998 and 2,500 poses for units 1 to 3 (two of unit 2's 3,000 frames had no detected face); the faces are an applied system of ours built for another purpose, and every quantity below is computed here and describes the reachable set and its projector, not a deployed controller. The supplementary video is a geometry illustration, not a learning result: it shows the babble sampling of one unit as it happens, with the output cloud filling in beside the moving face in the first two principal coordinates, and ends by overlaying the held-out human targets on the fitted box.

We estimate the reachable set as the image of a box under an isometry into the principal subspace,

$$\widehat{\mathcal{R}} = \{c + Uz : \ell \le z \le u\}, \quad U^\top U = I, \qquad \widehat{\Pi}(y) = c + U \operatorname{clip}\big(U^\top (y - c), \ell, u\big), \tag{27}$$

with c the sample mean, the columns of U the leading principal directions, and ℓ, u the 0.5th and 99.5th percentiles of the babble coefficients. Because U has orthonormal columns this *is* the Euclidean nearest point in $\widehat{\mathcal{R}}$: the box constrains principal coefficients, and clamping in ambient landmark coordinates would not be a projection onto the intersection. $\widehat{\mathcal{R}}$ is an estimated coefficient box, not a certified reachable set: actuator bounds alone do not make a nonlinear plant's output set box shaped, and the box approximates the sampled cloud rather than the plant's true boundary.

Table 31 reports three measurements of that estimate. Each cloud is captured by a few components, which describes the empirical output dimension of these plants and does not by itself make Theorem 2 inactive. Decomposing $\|y - \widehat{\Pi}(y)\|^2$ over held-out human targets separates coefficients driven past a fitted limit, where $\widehat{\Pi}$ is non-differentiable, from the component orthogonal to the principal subspace, where it is affine; the clamped part dominates on every unit, and on most held-out frames at least one fitted limit is binding, so the box is not a formality for this target distribution. What Table 31 does not measure is the projection-order bias: by Eq. 18 a limit can be binding on every frame with zero bias, and the $O(\sigma)$ term needs the conditional mean's distance to the limit on the scale of the conditional noise, which requires repeated observations per condition that these recordings do not contain.

Table 31: Measured geometry of three physical reachable sets: principal directions reaching a given cumulative realized variance, the split of $\|y - \widehat{\Pi}(y)\|^2$ over 200 held-out clips of video-derived mouth landmarks from two public audio-visual speech corpora, and how often a coefficient is at a bound.

	Components for			Split of $\|y - \widehat{\Pi}(y)\|^2$		Clamp active in	
Unit	95%	99%	99.5%	clamped	off-subspace	frames	coefficients
1 (design A)	2	5	6	88.2%	11.8%	92.8%	49.2%
2 (design B)	3	6	8	92.0%	8.0%	91.8%	39.6%
3 (design B)	2	5	6	81.9%	18.1%	85.8%	34.3%

What the measurement establishes is the geometry: a few principal directions, a coefficient box whose fitted limits bind on most target frames, and a projector that is affine except across those limits. That is the non-smooth type for which the clamp formula of Section 4 and Proposition 4 are written, and it is the case the smooth expansion of Eq. 9 does not cover; whether the bias and the metric gap are large on these plants depends on conditional quantities that Table 31 does not contain. This is one instance, we draw no ranking from it, and the box is an estimate; its role is to show that the geometric assumptions our analysis makes are not artefacts of simulated plants.

C.7 Public musculoskeletal control (MyoSuite)

We use MyoSuite (Caggiano et al., 2022) so that the projection operator, the plant, and every reported number rest on public components alone. A black-box cross-entropy-method optimizer (de Boer et al., 2005) projects desired output trajectories into muscle-command trajectories on a shared synergy basis. Every student emits muscle commands and executes on the impaired plant with no deployment inverse: "RAW" is behaviour cloning of the healthy expert's commands and uses no model of the impaired plant; PTF, CBD and oracle CBD are trained on the CEM inverse, computed offline through the impaired simulator, of the raw target, of the teacher's mean and of the clean target respectively.

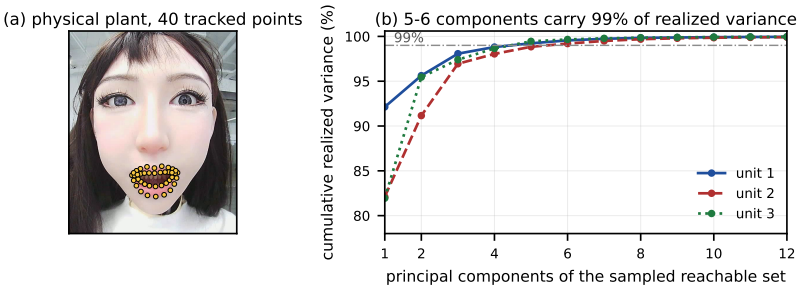


Figure 11: **A measured reachable set.** (a) One of the physical plants with the 40 tracked mouth landmarks that define the 80-D output space. (b) Cumulative realized variance of the babble-sampled output cloud on three units; the box of Eq. 27 is built on the components reaching 99.5% and is an estimate of the reachable set, not a certificate of it. A low empirical output dimension describes the geometry; by Section 3 it does not by itself make Theorem 2 inactive. The faces are an applied system of ours built for another purpose; the geometry reported here is measured in this work.

Black-box projection needs exact state restoration. A MyoSuite environment holds three MuJoCo model–data pairs, its own, an observation copy's and the robot's, and its own state accessor restores only the first, and only its positions and velocities. Evaluating candidates through that accessor is not reproducible: replaying one command sequence twice from a nominally restored snapshot moves the elbow by 0.96 rad, so the optimizer would be ranking candidates partly on simulator drift. Our snapshot stores the environment's state dictionary, the full MuJoCo integration state of all three pairs (including solver warm starts, controls and applied forces), the hidden fatigue variables and the target, and restoring it also resets the wrapper's step counter; random-number states are not part of it. A regression test requires two replays from one snapshot to agree, and they agree exactly, to the last bit, which is also what shows that the rollouts consume no random numbers the snapshot would have to carry. Targets are placed on each plant's measured output band rather than in joint-limit units, because a target family outside the achievable band collapses every arm onto the same failure and measures nothing.

Table 32: Public-simulator results, lower tracking MSE is better; every student emits commands, and "RAW" is healthy-command behaviour cloning (see the arm description above). MyoElbow is five seeds (mean ± std.) in this run and in the original run of the package; the success rate is this run's. MyoHand is one seed and shown only as a diagnostic.

	MyoElbow (five seeds)			MyoHand (one seed)	
Training target	MSE, this run	MSE, original	success	this run	original
Healthy-command BC ("RAW")	0.05956 ± 0.02240	0.06016 ± 0.02214	0.367	8.55×10^{-5}	8.22×10^{-5}
PTF	0.01869 ± 0.00427	0.01835 ± 0.00406	0.567	3.54×10^{-5}	3.39×10^{-5}
CBD	0.01815 ± 0.00408	0.01914 ± 0.00485	0.550	4.03×10^{-5}	4.17×10^{-5}
Oracle CBD	0.02475 ± 0.00952	0.02492 ± 0.00945	0.525	4.00×10^{-5}	4.02×10^{-5}

What MyoElbow tests. MyoElbow is an offline action-retargeting experiment, and its arms are not the three orders of the introduction (RAW, PTF and CBD as defined there) applied to one output target (Table 3). Every student emits muscle commands that the impaired plant executes, and the arms differ in the commands they are trained to imitate. The "RAW" arm is healthy-command behaviour cloning: it imitates the healthy expert's own commands, which the impaired plant executes to a different effect. PTF, CBD and oracle CBD imitate commands found by the offline CEM inverse on the impaired plant for, respectively, each noisy demonstrated output trajectory, the cross-fitted teacher's estimate of its conditional mean, and the clean trajectory. All four minimize command-space MSE, so the constrained Bayes rule, a statement about output-space loss, does not describe their training objective, and a nonlinear plant need not map the conditional mean of inverse commands to the projected conditional-mean output. The experiment therefore tests impairment-aware retargeting against healthy-command cloning, and within the retargeting recipe it compares per-sample inversion (PTF) with inversion of a teacher's mean (CBD); it is embodiment-transfer evidence of the kind

action-constrained imitation learning (Yeh et al., 2025) and planner-augmented policy distillation (Liu et al., 2022) provide, not a test of the output-space projection order.

On five-seed healthy-to-impaired MyoElbow (Table 32), learned CBD obtains tracking MSE 0.0182 ± 0.0041, PTF 0.0187 ± 0.0043, oracle CBD 0.0248 ± 0.0095 and healthy-command cloning 0.0596 ± 0.0224. Both retargeted orders beat healthy-command cloning in every seed, by 0.041 ± 0.008 and 0.041 ± 0.010 ($|t| \approx 4$–5, decided), a three-fold reduction in mean (per-seed ratios 2.6 to 3.9 and 2.1 to 5.4) that supports impairment-aware retargeting. Between CBD and PTF we did not resolve a difference: the paired difference is -0.0005 ± 0.0026 in this run and $+0.0008 \pm 0.0024$ in the original run, and the two arms swap places between the two runs while every other ordering is stable (Table 34). That oracle CBD trails learned CBD, by 0.0066 ± 0.0036 here and 0.0058 ± 0.0039 originally, worse in 8 of 10 seeds across the two runs and not decided in either, is not paradoxical. Proposition 3 is a statement about output-space targets and does not apply to command-space training. The two readings this leaves, that the inverse commands of the clean trajectory are harder for a small student to represent, or that they carry more of the inverse's own error, are both measured below (Table 33) and neither survives: the oracle student fits its cached commands better than the CBD student fits its own, in every seed, and its executed labels realize the clean trajectory at least as well; its deficit appears only on held-out contexts. We report the gap as not decided in sign and locate it in generalization across contexts, which these runs measure but do not explain. On MyoHand, the one-seed direct projector is useful and all three retargeted students improve tracking over healthy-command cloning, but PTF remains best; this is a scaling diagnostic, not statistical evidence.

What the arms are. Every MyoElbow student emits commands, so the experiment compares command-space training targets and not the output-space orders of Section 4; Table 3 and the paragraph below state the targets. Two experiments would connect it to the projection-order question and neither is run here: an output-emitting RAW arm that shares the inverse and the deployer with the retargeted arms, and a per-sample-versus-mean comparison within the retargeting recipe on more seeds.

The oracle-versus-learned CBD gap, measured on both sides. Two readings are consistent with the five-seed result, and both are qualitative because the arms train in command space and Proposition 3 does not describe them. Either the inverse commands of the clean trajectory are harder for a small student to represent than those of the smoother teacher-estimated trajectory, so that the student's approximation and optimization error grow by more than the target error falls; or the offline CEM inverse is itself in error with respect to the deployment plant, in which case the oracle arm pays that error in full while a biased teacher partially cancels it. The measurement that separates them is two-sided, and this revision runs both halves (Table 33): each arm's cached training commands are executed on the deployment plant and scored against the label's own target and against the clean trajectory, which isolates the inverse's error; and each student is scored against the cached commands it was trained on and on the training contexts it was trained from, which isolates the student's. Neither half favours the reading it tests. The oracle student's command-space fit is the better one (-0.016 ± 0.007 against learned CBD, lower in every seed), and the oracle's executed labels realize the clean trajectory as well as learned CBD's (-0.0009 ± 0.0006) and their own target about as well ($+0.0008 \pm 0.0006$); the offline projector and the deployment plant coincide in this run, so the executed and offline self scores agree to 10^{-9}. On the training contexts the two students track equally (-0.0008 ± 0.0013); the oracle student's deficit, $+0.0066 \pm 0.0036$, is a held-out-context effect. What the label of the clean trajectory changes is therefore not what the student can fit or what the inverse can realize, but how the fitted command map generalizes from 48 training contexts to 24 unseen ones. These runs locate the gap; they do not explain it, and it remains not decided in sign. Two candidates remain. A per-context inverse of the clean trajectory may be a less regular function of the context than the inverse of a teacher's mean. Or the arms differ in how many distinct targets they see: the oracle arm has one clean target per context (48), PTF one per demonstration (144) and learned CBD 83 to 88 distinct teacher trajectories, so the oracle student learns from the fewest and the least varied labels, which can hurt generalization to new contexts independently of what the labels are. An oracle arm given perturbed copies of its targets, matched in number to PTF's, would separate the two; these runs do not. What the two runs do establish is that the gap is not a fluke of one machine: it has the same sign and size in the original run and in the regeneration (Table 34).

Table 33: The two-sided attribution test on MyoElbow (this run, five seeds; mean ± std. per arm, then paired mean ± s.e. with the number of seeds in which the first arm is lower). Student tracking MSE on the held-out and on the training contexts; the student's command-space training loss against its cached commands; and the cached training commands executed on the deployment plant, scored against the label's own target (the noisy demonstration for PTF, the teacher's mean for CBD, the clean trajectory for oracle CBD and for the healthy commands) and against the clean trajectory.

	student tracking MSE		student fit	cached commands executed, MSE	
Training target	test contexts	training contexts	command MSE	vs. own target	vs. clean trajectory
Healthy-command BC	0.0596 ± 0.0224	0.0552 ± 0.0039	0.169 ± 0.009	0.0417 ± 0.0035	0.0417 ± 0.0035
PTF	0.0187 ± 0.0043	0.0172 ± 0.0037	0.224 ± 0.010	0.0116 ± 0.0030	0.0108 ± 0.0026
CBD	0.0181 ± 0.0041	0.0188 ± 0.0031	0.202 ± 0.011	0.0101 ± 0.0024	0.0117 ± 0.0023
Oracle CBD	0.0247 ± 0.0095	0.0180 ± 0.0023	0.186 ± 0.010	0.0109 ± 0.0014	0.0109 ± 0.0014
Oracle CBD − CBD	$+0.0066 \pm 0.0036$ (1/5)	-0.0008 ± 0.0013 (3/5)	-0.0161 ± 0.0071 (5/5)	$+0.0008 \pm 0.0006$ (1/5)	-0.0009 ± 0.0006 (4/5)
CBD − PTF	-0.0005 ± 0.0026 (3/5)	$+0.0016 \pm 0.0006$ (1/5)	-0.0215 ± 0.0068 (4/5)✓	-0.0015 ± 0.0006 (4/5)	$+0.0009 \pm 0.0005$ (1/5)
Oracle CBD − PTF	$+0.0061 \pm 0.0054$ (2/5)	$+0.0008 \pm 0.0016$ (2/5)	-0.0377 ± 0.0021 (5/5)✓	-0.0007 ± 0.0009 (3/5)	$+0.0000 \pm 0.0009$ (2/5)

Table 34: MyoElbow paired seed differences in tracking MSE (first minus second; mean ± s.e. and the number of seeds in which the first is better), in this run and in the original run.

Comparison	this run	original run
PTF − Healthy-command BC ("RAW")	-0.0409 ± 0.0083 (5/5)✓	-0.0418 ± 0.0084 (5/5)✓
CBD − Healthy-command BC ("RAW")	-0.0414 ± 0.0102 (5/5)✓	-0.0410 ± 0.0101 (5/5)✓
CBD − PTF	-0.0005 ± 0.0026 (3/5)	$+0.0008 \pm 0.0024$ (3/5)
Oracle CBD − CBD	$+0.0066 \pm 0.0036$ (1/5)	$+0.0058 \pm 0.0039$ (1/5)

MyoElbow uses six muscle actuators represented by a shared two-synergy search basis, an eight-step horizon, 48 training and 24 test contexts with three demonstrations each, demonstration noise 0.08, and a CEM projector with population 12, elite fraction 0.25, three iterations and initial standard deviation 0.7. The healthy expert supplies, for every context, a command sequence and the output trajectory it realizes on the healthy plant; the noisy demonstrations are those output trajectories plus smoothed noise. The four architecture-matched students (width 64, 350 epochs) map context and time to muscle commands and are trained by command-space MSE on: the healthy expert's commands (healthy-command behaviour cloning, the "RAW" row); the commands the CEM inverse finds on the impaired plant for each noisy demonstration (PTF); for the two-fold cross-fitted teacher's (width 64, 250 epochs) estimate of the conditional-mean trajectory (CBD); and for the clean expert trajectory (oracle CBD). All are evaluated by tracking MSE on the impaired plant at half strength. Table 35 gives the per-arm details of this run. Its two cost columns are what building the labels cost, which is not what this run spent: this run reused the projection cache that holds them, so the projection work it performed is zero for every arm, and the ledger is recovered from the cache itself, each entry recording the evaluations and the seconds of the projection stored in it (`scripts/myo_label_cost_provenance.py` writes it to `myoelbow/label_construction.csv`; a run that builds the labels records the same two quantities per row). Read that way the counts are deterministic, at 37 evaluations per projection throughout: PTF pays 144 projections per seed, one per demonstration, so 5,328 evaluations; oracle CBD pays 48, one per context, so 1,776; CBD pays 3,152 on average because only 83 to 88 of its 144 rows carry distinct cross-fitted teacher trajectories and the rest share a projection — its lower label cost is the reuse of projected teacher targets, not faster individual projection. RAW's entry is not zero either but shared: its labels are the healthy expert's commands, and that expert costs a further 2,664 evaluations per seed (72 projections, 4.4 s), which every arm's demonstrations derive from. The original run reports the same deterministic counts for PTF and oracle CBD and 3,249 for CBD, whose teacher targets differ from this run's at machine precision between the two PyTorch versions; its label seconds (87.3, 53.4, 29.5) are a slower machine's, and only the ratios carry over (PTF : CBD : oracle CBD $= 1 : 0.61 : 0.34$ there, $1 : 0.59 : 0.33$ here).

The fatigue screen crosses true initial fatigue and projector-assumed fatigue in $\{0, 0.35, 0.7\}$. The original run's remaining parameters are not recorded in the package, so the screen reported here was regenerated on the documented axes with the package's sweep defaults (24 training and 16

Table 35: MyoElbow per-arm detail for this run (five-seed means): success rate, final angle error (rad), what building that arm's labels cost per seed — simulator evaluations and seconds, recovered from the retained projection cache, since this run reused it and performed no projection of its own — student training time, and inference time per sample. RAW trains on the healthy expert's commands, whose own cost (2,664 evaluations, 4.4 s per seed) every arm shares through the demonstrations.

Training target	success	final angle err.	evals to build	build (s)	student (s)	inference (μs)
Healthy-command BC ("RAW")	0.367	0.337	shared	shared	0.59	11.3
PTF	0.567	0.190	5328	8.7	0.66	8.8
CBD	0.550	0.190	3152	5.1	0.61	8.0
Oracle CBD	0.525	0.223	1776	2.9	0.59	10.4

test contexts, two demonstrations per context, three seeds) and the CBD labels' CEM budget of the five-seed table (population 12, three iterations; Table 36). Healthy-command cloning and the teacher estimates are invariant to the projector setting by construction, with audit discrepancies of exactly zero; no method is uniformly best across the grid (CBD has the best mean in five cells, PTF in four); CBD beats healthy-command cloning in all nine; and oracle CBD has the worst mean in all nine. The search budget matters. At the sweep's default budget (population 8, two iterations) CBD beats healthy-command cloning in only five cells, and the least fatigued projector gives CBD's best mean at every true fatigue level, which would suggest that exact matching is not optimal over this short horizon; at the five-seed table's budget the most fatigued projector does at true fatigue 0.35 and 0.7 and the unfatigued one at 0, so that pattern belongs to the weaker search. Both screens are in the package. The attribution columns show where the mismatch enters: the executed error of CBD's cached commands equals their offline error on the diagonal and exceeds it by up to 0.07 in the corner cells, a direct reading of the offline projector's error on this plant.

Table 36: MyoElbow fatigue and projector-mismatch screen, regenerated (three seeds, sweep defaults, CEM population 12 with three iterations): tracking MSE per arm in each cell (best mean in bold) and, for CBD's cached commands, the executed error on the deployment plant against the error the offline projector reported for the same commands.

true	projector	tracking MSE, three-seed mean (best in bold)				CBD labels executed
fatigue	fatigue	healthy BC	PTF	CBD	oracle CBD	deployed / offline
0	0	0.096	0.067	**0.064**	0.131	0.0027 / 0.0027
	0.35	0.096	**0.082**	0.088	0.127	0.0110 / 0.0032
	0.7	0.096	0.107	**0.071**	0.159	0.0608 / 0.0217
0.35	0	0.092	0.057	**0.054**	0.123	0.0100 / 0.0027
	0.35	0.092	**0.068**	0.070	0.114	0.0032 / 0.0032
	0.7	0.092	0.088	**0.053**	0.141	0.0377 / 0.0217
0.7	0	0.113	**0.070**	0.071	0.137	0.0731 / 0.0027
	0.35	0.113	**0.073**	0.077	0.124	0.0446 / 0.0032
	0.7	0.113	0.066	**0.038**	0.117	0.0217 / 0.0217

MyoHand has a 115-dimensional observation, 39 muscle commands, a 15-dimensional fingertip output, and a common 14-group synergy basis, with a three-step horizon, 64 training and 16 test contexts, and width-128 students. The run has one seed and its tracking MSEs are 8.6, 3.5, 4.0 and 4.0×10^{-5} for healthy-command cloning, PTF, CBD and oracle CBD in this run (8.2, 3.4, 4.2, 4.0×10^{-5} originally). Direct black-box projection improves the diagnostic contexts, and distillation improves tracking relative to RAW, but it is insufficient for a paper-level claim about scaling.

D Additional investigations and negative results

Partial projection and projection-order paths. The static sweep carries a scalar path between the RAW and PTF targets, $(1-\alpha)y_i + \alpha\widehat{\Pi}(y_i)$ (Table 37). Every intermediate dose has a lower grand mean than either endpoint; the half dose is the lowest, level with learned CBD, and beats both endpoints in 14 of 36 cells (the quarter dose in 8, the three-quarter dose in 13). The static grid is one where PTF's bias is larger than RAW's capacity cost, so a partial projection buys variance reduction before the full bias arrives; we read this as a diagnostic of the two mechanisms' relative

size, not as a new target, and the cell counts are exploratory. Its practical appeal is real: it needs no teacher, no cross-fitting and no differentiable model, and here it is level with learned CBD. Its dose, however, is the best of three on the evaluation grid itself, so the grand mean flatters it by that selection; a practitioner would have to choose α on held-out data, which is what the cross-validated scalar selection audited below does. Audits of paths between the PTF and CBD targets, budget-matched stacking, cross-validated scalar selection, regional headroom and a smooth conditional gate, rerun here at $c = 2$, $\sigma = 0.4$ with the drivers' recorded parameters (five seeds), find what the original run found: the best scalar interior target improves on the better endpoint by 13–15% of its risk with the learned teacher (interior in four of five seeds) and by 1–6% when the teacher is replaced by the oracle conditional mean, and a context-conditional gate improves on CBD only when given oracle information (-1.5×10^{-4} at width 8, every seed) and not the raw information a practitioner has ($+0.5 \times 10^{-4}$, not decided). Both indicate shrinkage against teacher error, and neither is counted as a contribution.

Table 37: Partial PTF on the static grid: grand mean over 36 cells and five seeds under both criteria, and the number of cells in which the interior target beats both endpoints under task risk.

Target	excess task risk	D_B	beats both endpoints
RAW ($\alpha = 0$)	4.55×10^{-4}	5.04×10^{-4}	—
$\alpha = 0.25$	3.65×10^{-4}	4.04×10^{-4}	8/36
$\alpha = 0.5$	3.21×10^{-4}	3.65×10^{-4}	14/36
$\alpha = 0.75$	4.35×10^{-4}	5.06×10^{-4}	13/36
PTF ($\alpha = 1$)	5.25×10^{-4}	6.20×10^{-4}	—

Robust ensemble CBD. An ensemble over projector perturbations does not improve nominal mean performance anywhere we measured it: on the model-error screen it is worse than nominal CBD in 125 of 125 paired comparisons by $3.5 \times 10^{-3} \pm 0.1 \times 10^{-3}$, and on the trajectory plant it is worse at widths 16 and 32 ($+0.0032 \pm 0.0021$ and $+0.0019 \pm 0.0012$) and undecided at width 8. Robustness requires a principled ambiguity set and objective; averaging arbitrary projectors is not sufficient.

Pendulum. Pendulum has an analytically known one-dimensional action interval, so hard feasible output is the strongest control and CBD is unnecessary; the run confirms it (Table 38). An expert acting with torque limit 2 demonstrates with noise 0.6 to an imitator limited to 0.75; five seeds, 40 training episodes of 200 steps, 20 evaluation rollouts. The feasible-output student has the best return by a wide margin, PTF the worst; RAW has the lowest constrained-action MSE and yet a mediocre return, because its raw output violates the limit on 96% of steps and is only made feasible by the deployment clamp, and the two criteria order the arms differently. A feasibility-weighting arm, which down-weights infeasible demonstrations, is no better than RAW, and is not presented as positive external evidence.

Table 38: Pendulum, five seeds (mean $\pm$ std.): rollout return (higher is better), MSE between the deployed action and the clipped expert mean (lower is better), and the fraction of steps on which the student's raw output violated the imitator's limit before deployment clipping.

Method	return	constrained action MSE	violation before clip
RAW	-780 ± 55	0.0036 ± 0.0008	0.96
PTF	-856 ± 51	0.0120 ± 0.0024	0.35
CBD	-762 ± 64	0.0086 ± 0.0007	0.47
Oracle CBD	-752 ± 105	0.0105 ± 0.0010	0.46
Feasible (RAW targets)	-660 ± 55	0.0057 ± 0.0033	0.00
Feasibility weighting	-802 ± 42	0.0058 ± 0.0009	0.94

E INDEPENDENT RE-IMPLEMENTATION

What it shows. Before the released package was audited, we wrote a second implementation of the static benchmark from the paper's description alone. Its projector agrees with the package's to

5.4×10^{-7}, its metrics are computed by different code, and its sweep is larger: 13 arms, 36 cells, and a 2×2 grid of training-set size $\{512, 2048\}$ and optimization budget $\{600, 3000\}$ epochs, 9,360 rows in all. It is a different *instance* of the same problem, and we report it as a stress test of the paper's thesis and not as a controlled factor: the conditional mean is displaced vertically, by 0.10 to 0.60, rather than along the curve's normal by 0.12 to 0.27; the noise is isotropic and homoscedastic; the inputs span a wider range; the curvatures are $c \in \{1, 2, 4\}$ rather than $\{0.5, 1, 2\}$; and the students are trained with restarts on a fixed budget. Its curvature parameter is $\kappa = c/2$ and its metrics sum the two coordinates, and every comparison below is made at the same c and per coordinate. On that instance RAW wins 31 to 36 of the 36 cells at every point of the protocol grid and PTF is behind RAW at every curvature, by $2.3 \times 10^{-4} \pm 0.5 \times 10^{-4}$ pooled at $n = 2{,}048$ and 3,000 epochs ($|t| = 4.4$; decided at $c = 1$ and $c = 4$ separately). The population projection-order discrepancy has to be counted after deployment to compare the instances. Integrated at the same curvature, it is 1.3 to 3.3 times *larger* on that instance than on the package's, but most of it is normal to the curve, and after the deployment projection every output student passes through, what remains is *smaller* there in 7 of the 9 matched cells (Table 10; Appendix A): that instance leaves CBD less to remove. What differs is what CBD's recipe adds on that instance, read through two intervention contrasts and not as measurements of the terms of Proposition 3. Replacing the learned teacher by the oracle is worth 2.1–2.4×10^{-4} at every curvature there ($|t| \geq 7$), against 0.6×10^{-4} at $c = 0.5$, 0.4×10^{-4} at $c = 1$ and the opposite sign at $c = 2$ on the package's instance; and the projected target itself is harder to fit than the raw mean: under the 600-epoch budget even oracle CBD loses to RAW at every curvature, and under 3,000 epochs the oracle pulls ahead at $c \leq 2$ and still loses at $c = 4$. Standard distillation, which caches the teacher's output without projecting it, has the lower mean than CBD on that instance at every curvature ($|t| = 2.8$, 1.4 and 6.9 at $c = 1, 2, 4$, decided at $c = 4$); on the package's it does so at $c = 1$ and loses at $c = 2$. Which of the instance's differences drives the larger oracle-minus-learned contrast these runs cannot say, and we do not attribute it: the noise structure, the offset, the input range and the teacher recipe changed together. We report the pair because it is the paper's thesis in one sentence: the order that wins is a property of the joint law of (X, Y), the geometry of $\mathcal{R}$ and the terms of Eq. 10 a given recipe incurs, not of the order, and a benchmark that does not state its noise structure and its teacher has not stated its result. The same re-implementation also measured a failure mode worth knowing: under a fixed 3,000-epoch budget at $n = 512$, every teacher width over-fits the observation noise after about 500 epochs, its conditional-mean error then rises monotonically and faster for wider teachers, and held-out early stopping cannot rescue it at this signal-to-noise ratio (Table 41). CBD pays for that twice, once in the teacher and once in the student, and RAW once, so an unexamined budget is a bias toward RAW.

What was built. A second implementation of the static benchmark was written from the paper's description before the released package was audited, with its own closed-form projector (enumeration of the real roots of the cubic stationarity condition of the quartic squared distance, together with the two endpoints), its own students, teachers and metrics, and a larger protocol: 13 arms (the three orders, oracle CBD, standard distillation, three doses of a projected partial-distillation control $\widehat{\Pi}((1-\alpha)y_i + \alpha T^{-k(i)}(x_i))$, which is not the partial PTF of the main sweep, a soft constraint, and feasible-output students on four targets) over 36 cells of width, curvature and noise, crossed with training size $\{512, 2{,}048\}$ and optimization budget $\{600, 3{,}000\}$ epochs, with three restarts per student selected on training loss; 9,360 seed-level rows. The two projectors were compared on 4,000 points at each of three curvatures: the largest discrepancy is 5.4×10^{-7} and the closed-form projector never finds a nearer point than the package's grid-and-Newton one.

What differs. The package displaces the conditional mean from the curve along its normal by an input-dependent amount and draws anisotropic, correlated, input-dependent noise, a design its source comments explain as a guard against projection and expectation commuting by symmetry. The re-implementation, written from the text, displaced the mean vertically by 0.10 to 0.60 and used isotropic homoscedastic noise; its inputs are uniform on $[-1.5, 1.5]$ rather than $[-1.25, 1.25]$, its curvatures are $c \in \{1, 2, 4\}$, its students use three restarts selected on training loss and a fixed epoch budget without patience, and its teacher is a five-fold cross-fitted ensemble. The two share the parabola family and the student class. Because several things differ at once, the comparison is a different-instance stress test and not a factorial isolation of the noise structure.

Table 39: Independent re-implementation over its protocol grid ($c \in \{1, 2, 4\}$, five seeds, per-coordinate units): cells won by RAW/PTF/CBD among the three orders under excess task risk and under D_B, cells decided by a 95% paired t-test, cells whose winner changes between criteria, the pooled teacher cost of CBD (learned minus oracle CBD, task risk), and the pooled difference between standard distillation (KD, the teacher's output cached without projection) and CBD (negative: distillation better; the count is the number of the 180 cell-seed pairs in which distillation is better).

n	epochs	task risk	decided	D_B	flips	CBD − oracle	KD − CBD (KD better)
512	600	31/4/1	12	32/4/0	5	4.97×10^{-4}	-3.11×10^{-4} (141)
512	3000	34/1/1	12	35/1/0	1	9.00×10^{-4}	-2.50×10^{-4} (145)
2048	600	36/0/0	17	36/0/0	0	1.72×10^{-4}	-4.65×10^{-4} (154)
2048	3000	34/2/0	19	34/1/1	2	2.23×10^{-4}	-1.89×10^{-4} (158)

Units and alignment. The re-implementation's result table stores excess task risk and D_B as the *sum* over the two output coordinates, whereas the package stores per-coordinate means; every number below and in Table 39 has been divided by two. Its curvature parameter is $\kappa = c/2$, so its sweep covers $c \in \{1, 2, 4\}$ and overlaps the package's grid at $c = 1$ and $c = 2$ only; the discrepancy column of Table 10 is computed for it at $c \in \{0.5, 1, 2\}$ from its own geometry module by the same quadrature as for the package. An earlier version of this comparison placed κ and c side by side and used the Monte-Carlo plug-in; both errors are corrected here.

Numbers behind the summary. RAW wins 31 to 36 of 36 cells under task risk at every point of the protocol grid (Table 39), PTF at most 4 and learned CBD at most 1, and neither more data nor a longer budget moves it; PTF is behind RAW at every curvature ($|t| = 3.3$, 2.6 and 7.2 at $c = 1, 2, 4$ under $n = 2{,}048$ and $3{,}000$ epochs, decided at $c = 1$ and $c = 4$; pooled $|t| = 4.4$); the population discrepancy of Table 10 is 1.3 to 3.3 times larger here before the deployment projection and smaller in 7 of 9 matched cells after it. Under the 600-epoch budget oracle CBD loses to RAW at every curvature ($+1.0$, $+2.5$ and $+6.4 \times 10^{-4}$ at $c = 1, 2, 4$, $n = 2{,}048$), so the projected target is harder for the student to fit than the raw mean; under $3{,}000$ epochs the oracle pulls ahead at the two lower curvatures (-1.3 and -1.7×10^{-4}) and still loses at $c = 4$ ($+2.1 \times 10^{-4}$), which is beyond the package's curvature range. The teacher cost, learned minus oracle CBD, is 2.1–2.4×10^{-4} at every curvature under the long budget ($|t| \geq 7$) and 1.5–2.1×10^{-4} under the short one, against 0.6×10^{-4} at $c = 0.5$, 0.4×10^{-4} at $c = 1$ and -0.4×10^{-4} at $c = 2$ on the package's fifteen-seed grid; it falls with sample size as the theory predicts, from 9.0×10^{-4} at $n = 512$ to 2.2×10^{-4} at $n = 2{,}048$, but never below the projection step's gain. Standard distillation, which caches the teacher's output without projecting it, beats CBD here in 48 to 58 of 60 pairs at every curvature ($|t| = 2.8$, 1.4 and 6.9; decided at $c = 4$, and at $c = 1$ short of the threshold by 0.01); on the package's instance it wins at $c = 1$ and loses at $c = 2$, where the discrepancy of Table 10 is largest. The projection step of CBD therefore earns its place only where the part of the population discrepancy that survives deployment is large enough to pay for the teacher error and the harder target it rides on. Which of the instance's differences drives the larger teacher term, the isotropic noise, which puts the full σ on the coordinate along the curve where the package puts 0.55–$0.9\,\sigma$, the larger vertical offset, the wider input range, or the re-implementation's teacher recipe, these runs do not separate, and we make no causal attribution; the budget measurement below shows the recipe is at least part of it. We regard the pair of instances as a single result: the regime map is a function of the instance and of the terms of Eq. 10 a recipe incurs, and a benchmark for training order must report both.

Restart selection. The re-implementation trains every student with three restarts and keeps the one with the lowest training loss. Until this version the loss it compared was the one computed in the last epoch *before* that epoch's parameter update, while the network it kept was the updated one, the same one-step mismatch that the package's training loop had before its correction. The selection now scores the parameters it keeps, and the protocol grid was rerun; Table 40 measures the effect against the retained earlier table. The three-order winner agrees in 143 of the 144 cells over the four protocol points, the pooled PTF-against-RAW, oracle CBD against CBD and distillation against CBD contrasts change by at most 6% and keep their verdicts, and the cell means of the three orders, oracle CBD and distillation change by a median of 0.01% to 1.3% depending on the protocol point, with a

90th percentile of at most 7.5% and one cell at 60% at $n = 2{,}048$, 3,000 epochs. Every number in this appendix and in Appendix E is from the rerun.

Table 40: Effect of the restart-selection correction in the re-implementation, per protocol point: cells whose three-order winner agrees between the earlier table and the rerun, the distribution over cells and arms (three orders, oracle CBD, distillation) of the relative change of five-seed cell means, and the pooled PTF-against-RAW contrast (mean, $|t|$) before and after.

n	epochs	winners agree	median rel. diff.	90th pct.	max	PTF − RAW before ($\lvert t\rvert$)	after ($\lvert t\rvert$)
512	600	36/36	6.5×10^{-7}	0.5%	4.3%	$+2.68 \times 10^{-4}$ (2.3)	$+2.68 \times 10^{-4}$ (2.3)
512	3000	35/36	7.8×10^{-3}	5.9%	19.3%	$+3.42 \times 10^{-4}$ (4.4)	$+3.36 \times 10^{-4}$ (4.0)
2048	600	36/36	9.0×10^{-7}	0.8%	5.0%	$+3.67 \times 10^{-4}$ (1.8)	$+3.66 \times 10^{-4}$ (1.8)
2048	3000	36/36	1.3×10^{-2}	7.5%	60.1%	$+2.31 \times 10^{-4}$ (4.4)	$+2.32 \times 10^{-4}$ (4.4)

Optimization budget as an axis. The re-implementation's first teacher sweep pointed the wrong way, with conditional-mean error rising from 0.0058 to 0.0158 as teacher width went from 16 to 64 at $n = 512$. Measuring the training curve explained it (Table 41): under a fixed 3,000-epoch budget every width reaches its best conditional-mean error between epochs 500 and 1,000 and then fits the observation noise, faster for wider teachers. Held-out early stopping did not rescue it: with 128 validation points the signal to resolve (≈ 0.002) sits below the noise floor ($2\sigma^2 = 0.32$), three splits chose epochs 100, 100 and 350, and refitting at the chosen epoch gave 0.0052 at width 64 against 0.0019 at a fixed 500. No weight decay between 10^{-4} and 10^{-2} matched a fixed 500-epoch budget at every width. The re-implementation therefore made the budget an experimental axis rather than a guess, which is the 2×2 protocol grid above; the package's 700-epoch students and patience-on-training-loss teachers sit on the safe side of the same curve. The asymmetry matters for the paper's question: an over-trained teacher costs CBD twice, in the teacher and again in the student that copies it, and RAW once.

Table 41: Conditional-mean test error of a teacher against training epochs at $n = 512$, $\sigma = 0.4$ (independent re-implementation, single data set): every width bottoms out between 500 and 1,000 epochs and rises afterwards.

epochs	width 16	width 32	width 64
500	0.00274	0.00194	0.00189
1000	0.00205	0.00225	0.00216
2000	0.00234	0.00350	0.00348
3000	0.00402	0.00449	0.00563

F Reproducibility statement

Every number in this paper is regenerable from released material: the linear, curved, trajectory, teacher, model-error, pendulum and MyoSuite experiments live in one anonymized package with an entry point per suite, each run writes one row per (arm, cell, seed) carrying both metrics of Appendix C.1, and the tables and figures in this paper are generated from those rows by two scripts rather than transcribed. The numbers reported are those of a full rerun of the package on a second machine; seeds are explicit and are 0–4 unless stated otherwise. The original run used Python 3.11.9, PyTorch 2.10.0 (CPU), NumPy 2.2, MuJoCo 3.6.0 and MyoSuite 2.12.2; the rerun used the same versions except PyTorch 2.6.0 (CPU).

Agreement between runs. Table 42 compares the static grid on seeds 0–4, restricted to the four arms whose student initialization is the same in every run (RAW, PTF, CBD and standard distillation; inserting the cross-fitted arm shifted the initialization seeds of the arms after it). Against the original run, the three-order winner agrees in 35 of 36 cells (the exception, width 32, $c = 2$, $\sigma = 0.05$, is not decided), and the per-cell five-seed means agree to a median relative difference of 0.2%, with a 90th percentile of 1.8% and a maximum of 20% in that cell. Against the run preceding the checkpoint correction, which is otherwise identical code on identical seeds, the winner agrees in 34

of 36 cells (both exceptions not decided), cell means differ by a median of 0.2% (90th percentile 1.1%, maximum 10%), and the 720 individual rows differ by a median of 0.1%, with 83% of rows within 1%, 99.9% within 10% and one row at 40%; every row ran the same number of epochs in both runs, as the correction requires. The trajectory comparison with the original run is made on the hard-switch plant, which is the plant that run used (Table 24): on seeds 0–4 the orderings of the four target-based arms agree with the original run at widths 8 and 16 under realized target MSE and differ only by swaps of arms that are not decided (CBD and oracle CBD at width 32; RAW and oracle CBD at width 8, and CBD and oracle CBD at widths 16 and 32, under D_B). MyoElbow agrees within one standard deviation for every arm (Table 32), with the same paired-difference signs (Table 34); the one-seed MyoHand values agree to 4%.

Table 42: Static-grid agreement between runs on seeds 0–4: the original run against this run, and the run preceding the checkpoint correction against this run. Cells whose three-order winner agrees, and the distribution over cells and arms of the relative difference of five-seed means.

Comparison	winners agree	median rel. diff.	90th pct.	max
original run vs. this run (seeds 0–4)	35/36	2.3×10^{-3}	1.8%	19.7%
checkpoint after vs. before the step (seeds 0–4)	34/36	1.7×10^{-3}	1.1%	10.1%

How the sweeps were run. The static and trajectory drivers are single-process; the regeneration sharded them by (seed, curvature, noise) and (seed, dead-zone error, coupling error) and ran the shards in parallel. Because the driver re-seeds every random generator at the top of each such cell, the sharded run is bit-identical to a single-process run. The package sets the BLAS and OpenMP thread variables to one when it is imported, unless they are already set, because side-by-side shards with the libraries' default of one thread per core run an order of magnitude slower than the serial run. The drivers ship with the package. The static grid of this version contains $16{,}200$ rows over fifteen seeds, of which seeds 0–4 correspond to the $5{,}220$ rows of the original run plus the cross-fitted CBD arm; the trajectory grid contains $4{,}815$ rows over fifteen seeds (eleven arms on the 3×3 model-error grid plus the two warm-started, the two long-budget cold and the four compute-matched controls on the exact plant), of which seeds 0–4 correspond to the original $1{,}350$ plus that arm and those eight controls, now on the continuous plant; the two-link arm grid contains $1{,}560$ rows over thirty seeds (thirteen arms, two regimes, two widths) and its pilot 260 rows over five other seeds, used only for the sample size; the model-error screen contains the same 500 rows as the original. The teacher sweep, pendulum and projector validation had no retained configuration in the original run and were run at the settings stated in this appendix.

Provenance of the reported rows. The static grid, the teacher sweeps, the model-error screen, the pendulum and the MyoSuite rows are those of the rerun; everything that uses the trajectory plant was run again after its rate switch was made continuous (the trajectory grid, the projector validation and the reachable-residual check), with the hard-switch plant kept only as the diagnostic of Table 24; and the exact-plant trajectory table was run once more, in full, when the two long-budget cold arms and the imitation-stage scoring of the warm arms were added, every earlier row of it reproducing bit for bit. Corrections made along the way during internal review, all visible in the package's history, are: the training loop and the trajectory projector returned the iterate one step after the one whose loss they recorded (Table 42 measures the effect); the projection-order discrepancy was a Monte-Carlo plug-in with a variance floor (Table 11); teacher time had been copied into every arm's cost; pooled uncertainties had treated the cells of one seed as independent; the deployment inverse was counted as 121 rather than 122 plant calls, and the feasible student's single deployment realization as none (the ledger now counts forward calls on both sides); the raw-target warm start had not been charged its inverse; the cost panel of the trajectory figure had been drawn from the preceding run's rows while its tables used the current run's (every figure is now regenerated from the same rows as the tables); and the re-implementation's restart selection scored the iterate before its last update (Table 40). Later in that review the exact-plant trajectory table was run a third time, to time the feasible student's single deployment realization, every accuracy column again reproducing bit for bit; the MyoElbow five-seed table was run again to add the two-sided attribution test of Appendix C.7, its earlier columns reproducing bit for bit; the fatigue screen was regenerated on this machine with the package's sweep defaults, and later again at the five-seed table's CEM budget, which the paper reports; and the audit chain of Appendix D was rerun with the drivers' recorded parameters. Later still the exact-plant

trajectory table was run a fourth time, in full, to add the four compute-matched controls of Table 20, whose design was written into the package before the run; every accuracy column of the fifteen earlier arms reproduces bit for bit, and every wall-clock of the table comes from this run. The two-link arm grid of Table 22 was then run once, pilot and confirmatory run, with its design written into the package before the pilot. Last, the released driver ran the warm start with a misspecified embedded plant (1,440 rows, design written into the package before the run, its cold rows reproducing the grid's bit for bit), the output students again with their predictions recorded before the inverse (180 rows, every earlier column reproducing bit for bit), and the fatigue screen at the larger CEM budget; the quadrature gained its after-deployment column, its other columns unchanged. Every experiment reported in this paper comes from that chain of runs.

Three checks are part of the package. The curved benchmark's projector is compared against an independent closed-form root enumeration (Appendix E), so its exactness is measured, not assumed. The MyoSuite state snapshot is covered by a regression test that replays one command sequence twice from a restored snapshot and requires bit-identical output; it is the test that exposed the second, unrestored state object. And the identity relating the two metrics (Eq. 24) is verified on the fitted models themselves, since both sides are recorded for every row. A regression test checks that the training loop and the trajectory projector return the iterate whose loss they recorded, and another that the quadrature estimator agrees with a large Monte-Carlo sample and exposes the floor of the 256-draw plug-in, one pins the stop-crossing discontinuity of the two-link arm described above, and one pins the two MyoElbow cost ledgers: a projection served from the cache must add nothing to the work a run performs while keeping the cost of building it, and a projection shared by several rows must be charged once. Two more pin the trajectory ledgers the main text quotes: the offline ledger must exclude exactly the deployment term of Table 4, and the cold students' ratio to RAW is aggregated per width, since it is not the same at each. Others drive the from-scratch driver through failed shards, incomplete merges and a resume without running an experiment, and check that the aggregation stops on a named input that does not exist instead of reading the shipped rows. The package's 101 unit and regression tests pass.

End-to-end rerun. The released driver was run once from scratch, with nothing copied from an earlier run: every experiment here, the MyoSuite projection caches and the independent re-implementation's grid included, was regenerated in one pass (6 h 15 min on twelve cores), and the aggregation was then pointed at that directory. Of the 47,094 released quantities, 44,662 come back unchanged, 2,384 differ, and 48 have no counterpart, being the MyoElbow label-construction ledger that a run building its own labels records inline instead. Every difference is a wall-clock time or a ratio of them (2,295), a counter of work the released run did not perform because its labels were cached (7; the rerun builds them and pays the 5,328, 3,152 and 1,776 evaluations per seed of Table 35, an independent check of that ledger), an accounting field older hard-switch rows predate (24, none an accuracy column), one teacher diagnostic's recording granularity (36) with the ratios formed from it (20), which agree to seven significant figures, or a path (2). No reported accuracy, statistic, decision or count differs.

The linear-case sweep of Table 2 is closed form. The physical geometry of Appendix C.6 is computed by a released script from the babble recordings, whose acquisition protocol that appendix states. Simulator binaries and caches are not redistributed; they are reconstructible from the public versions above. Machine-specific paths are not part of the specification and do not appear in the archive.

What the supplementary archive contains. The anonymous archive submitted with this paper holds the benchmark package with its tests and the drivers used here; the seed-level rows of every experiment in this paper and of the original run, from which the archive's own aggregation regenerates, without rerunning anything, the experimental numbers, the generated tables of this PDF byte for byte and its synthetic figures; the independent re-implementation of Appendix E with its rows and the probe behind Table 41; and, for Appendix C.6, the babble-sampled point clouds of the three units together with the 200 held-out target clips as 80-dimensional mouth-landmark features, from which Table 31 and the data panels of Figures 1 and 11 recompute exactly (their photographs are not included). Its README maps every table and figure to the script that produces it. The targets are landmark features extracted from the test splits of two public audio-visual speech corpora and are redistributed only in that derived form under those corpora's licences; no video, audio or identity is included.

G BROADER IMPACT

Reachability-aware supervision could reduce unsafe or mechanically impossible commands in assistive devices, musculoskeletal controllers and expressive robots, and could equally create false confidence where the reachable-set model omits rare failures, human contact, wear or distribution shift; it should complement runtime safety monitors and hardware validation rather than replace them. The two deployment forms studied here fail differently. An output student deployed through an inverse consults the plant model at every query, so a re-identified model can be swapped in at run time, at the price of the inverse's latency; a command-emitting student carries the model it was trained through in its weights, runs open loop, and must be retrained when the plant changes, and its error grows quickly with model error (Figures 7 and 10). Public musculoskeletal simulations avoid human-subject data in the present study; deployment to rehabilitation or medical systems would require domain-specific validation and oversight.

H EXTENDED RELATED WORK

This section gives the comparison that Section 6 summarizes.

Feasibility-aware imitation learning filters or weights demonstrations from agents with different dynamics (Lee et al., 2020; Cao et al., 2022), out-of-dynamics imitation models multimodal demonstrators (Qiu et al., 2023), retargeting constructs executable motions before policy learning, and action-constrained imitation learning (Yeh et al., 2025) builds a feasible surrogate data set by model predictive control for an imitator with a smaller action set. What these align is a trajectory in time; our object is the output-space target and the order in which averaging and projection are taken. In those terms a feasible surrogate data set is the PTF order applied per demonstration, and feasibility weights pull the RAW target toward the demonstrations nearest the reachable set (no better than RAW on our pendulum, Appendix D); we compare the orders, not those pipelines. Optimizer-generated supervision is established: guided policy search trains policies on trajectory-optimization solutions (Levine & Koltun, 2013) and planner-augmented policies are distilled into planner-free ones (Liu et al., 2022); Section 5.2 reuses commands a projected-target pipeline has already computed, even where its projected outputs are the worse labels.

Gupta et al. (2026) derive controllers for noncuspidal $3R$ chains under which a demonstrated policy transfers with certified feasibility, the exact instance of EMBED where an infeasible task can be refused. Our targets are unreachable for every candidate output, so the object of interest is $\Pi_{\mathcal{R}}(\mathbb{E}[Y \mid X])$ rather than a certified halt, and the reachable set is known only through a model or samples. Architectures with built-in output constraints (Amos & Kolter, 2017; Agrawal et al., 2019; Brosowsky et al., 2021) are the same order and our strongest control; our results favour them wherever their model is accurate and the search through it starts in the right basin. Extrinsic regression (Lin et al., 2017) projects an ambient conditional mean onto a manifold and Fréchet regression (Petersen & Müller, 2019) defines the conditional mean on an object space, but in both the responses already lie on the set, whereas ours do not and the deployment student has finite capacity; reduced-rank regression (Anderson, 1951; Izenman, 1975) is the classical account of rank-constrained prediction, of which Theorem 2 is the deployment-projected reading; decision-focused learning (Donti et al., 2017; Elmachtoub & Grigas, 2022) changes the criterion, where we keep squared realized-output loss and change the target; and CBD differs from knowledge distillation (Hinton et al., 2015) and logic-rule distillation (Hu et al., 2016) in its defining object, the constrained Bayes target. This paper contributes the training-order theory and claims no hardware measurement of a deployed learned controller.