CALIBRATED CANDIDATE REPLACEMENT UNDER IMPERFECT WORLD MODELS VIA BASELINE-RELATIVE LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

World-model errors can make candidate scores unreliable. A higher predicted return does not always justify replacing the agent's incumbent action. We propose *Baseline-Relative Value Learning* (BRVL), a post-training selector for frozen world-model agents. BRVL learns candidate–incumbent return differences from paired offline simulator rollouts under a common baseline continuation. Its objective combines relative-return regression with a candidate-set loss for forgone gains and harmful replacements. At deployment, BRVL replaces the incumbent only when an alternative has a positive calibrated lower score. We evaluate BRVL with DreamerV3 and TD-MPC2 on DMControl, Meta-World, and ManiSkill2. BRVL improves mean closed-loop performance and frozen-set net gain over matched-supervision regression baselines in all six settings. The gain–harm trade-off varies across settings.

1 INTRODUCTION

World-model agents use learned dynamics for imagined policy learning and online planning (Ha & Schmidhuber, 2018; Hafner et al., 2020; Schrittwieser et al., 2020). Candidate scores inherit errors from predicted dynamics, rewards, and terminal values (Feinberg et al., 2018; Chua et al., 2018; Asadi et al., 2019). A higher predicted return does not always justify replacing the *incumbent*, the action the agent would otherwise execute. We study selection from state-dependent candidate sets with the pretrained agent fixed. Comparison errors have different decision costs (Farahmand, 2011; Lazaric et al., 2016). Our goal is to capture useful improvements and limit harmful-replacement loss, not to recover complete candidate rankings.

Advantage estimation, value–advantage decomposition, and direct advantage learning provide relative representations (Schulman et al., 2015b; Wang et al., 2016; Pan et al., 2022; Pan & Schölkopf, 2026). Relative Value Learning estimates differences between state values (Höftmann et al., 2026). We instead compare candidate and incumbent actions at the same state under a common continuation policy. Conservative improvement and selective prediction motivate retaining baseline behavior when evidence is weak (Petrik et al., 2016; Laroche et al., 2019; Geifman & El-Yaniv, 2017). Recent work also studies calibration for control and decision risk (Prinster et al., 2026; Yeh et al., 2025). The remaining challenge is to turn noisy relative predictions into a replacement decision. This requires accounting for comparison error across the candidate set.

We propose *Baseline-Relative Value Learning* (BRVL). It combines paired relative supervision, candidate-set decision-aware learning, and blockwise calibration. Offline simulator rollouts compare candidate and incumbent actions from the same restored environment and agent state. Both branches then follow the frozen baseline policy. A relative predictor learns the return differences through regression and a decision cost for forgone gains and harmful replacements (Elmachtoub & Grigas, 2022; Yamao et al., 2026). Split-conformal calibration (Angelopoulos & Bates, 2023) calibrates prediction errors against finite-rollout measurements across all candidates and roots in a source block. BRVL replaces the incumbent only when a calibrated lower score is positive. Deployment uses agent-derived features without counterfactual environment branching or measured candidate returns.

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.

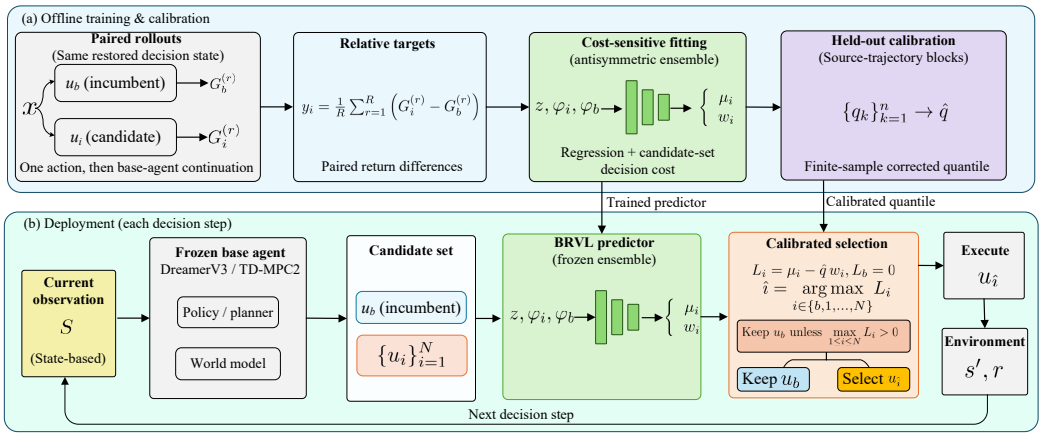


Figure 1: **Overview of BRVL.** (a) Offline fitting and blockwise calibration. (b) BRVL executes the alternative with the largest positive $L_i = \mu_i - \widehat{q} w_i$, or retains the incumbent if none exists. The base agent remains fixed.

Under source-block exchangeability, BRVL gives simultaneous coverage of fresh paired return measurements. With conditionally unbiased measurements and a known return bound, this also controls marginal expected harmful loss. We relate baseline-relative values to closed-loop return through baseline advantages. Experiments with DreamerV3 and TD-MPC2 (Hafner et al., 2025; Hansen et al., 2024) measure frozen net gain, measured harmful loss, and complete-episode performance. Matched-supervision baselines, threshold diagnostics, and component ablations assess decision-aware learning and calibrated fallback.

2 Related Work

World models support trajectory-sampling planning (Chua et al., 2018) and model-based value expansion (Feinberg et al., 2018; Buckman et al., 2018). However, longer rollouts and more accurate dynamics can yield diminishing gains in value expansion (Palenicek et al., 2026), while search can degrade performance despite accurate models (Chang et al., 2026). BMPC and BOOM improve planner–policy interaction through imitation and alignment (Wang et al., 2025; Zhan et al., 2025). BRVL focuses on post-training candidate selection without updating the base agent.

Direct Advantage Estimation learns advantages directly (Pan et al., 2022; Pan & Schölkopf, 2026). RVL learns antisymmetric state-value differences through pairwise Bellman updates (Höftmann et al., 2026). BRVL learns same-state candidate–incumbent return differences from paired rollouts under a common baseline continuation. With single-step execution and a deterministic baseline, this target equals the baseline advantage. Its candidate-set objective penalizes forgone gains and harmful replacements. This design draws on action-loss analysis and decision-focused learning (Farahmand, 2011; Elmachtoub & Grigas, 2022).

SPIBB constrains deviations from a baseline (Laroche et al., 2019). Selective prediction studies risk–coverage trade-offs (Geifman & El-Yaniv, 2017). Conformal methods address risk control and training (Angelopoulos et al., 2024; Yeh et al., 2025), trajectory filtering (Nguyen et al., 2026), and reference-policy calibration (Prinster et al., 2026). BRVL calibrates a frozen cost-sensitive predictor to obtain simultaneous lower bounds on paired return-difference measurements. It uses these bounds to choose between candidate replacement and incumbent fallback.

3 Method

BRVL separates comparison learning from replacement decisions. Paired offline rollouts train a decision-aware ensemble. Held-out source blocks then set its calibration margin (Fig. 1(a)). At deployment, the frozen agent and fixed generator provide the incumbent and alternatives (Fig. 1(b)). BRVL compares their calibrated lower scores and replaces the incumbent only when an alternative has a positive score.

3.1 FINITE-CANDIDATE REPLACEMENT

Let x be a recoverable decision state. It includes the environment state and the history-dependent agent or planner carry. Let z be the frozen agent's deployable representation. A fixed generator produces $\mathcal{C}(x) = \{u_b, u_1, \ldots, u_N\}$, indexed by $\mathcal{I} = \{b, 1, \ldots, N\}$. The incumbent u_b is the action proposed by the frozen base agent π_b. Each selected action executes for one decision interval with the configured action repeat, so $H_{\text{exec}} = 1$. For supervision and frozen-set evaluation, branches then follow π_b to the return endpoint. These branches measure the effect of changing the initial action under a common continuation. Closed-loop deployment instead reapplies BRVL at each decision. The horizon H_{plan} controls imagined candidate evaluation.

Conditional on the generated candidates, define

$$J(x,u) = Q^{\pi_b}(x,u) = \mathbb{E}\left[\sum_{t=0}^{T_{\text{end}}-1} \gamma^t r_t \,\middle|\, x, u_0 = u, \pi_b \text{ thereafter}\right], \qquad \Delta_i(x) = J(x,u_i) - J(x,u_b), \qquad \Delta_b(x) = 0. \tag{1}$$

Rewards and discounts follow the agent's decision-time convention. T_{end} is the remaining task horizon, subject to the task's termination and timeout rules. All returns, including $J(x,u)$ and $G_i^{(r)}$, use a common positive task scale fixed from fitting data. Selecting $\hat{i}$ incurs candidate-set regret $\max_{i\in\mathcal{I}} \Delta_i - \Delta_{\hat{i}}$ and harmful-replacement loss $[-\Delta_{\hat{i}}]_+$, where $[a]_+ = \max(a, 0)$. Retaining the incumbent gives zero improvement. Simulator information used only for restoration is excluded from selector inputs.

3.2 PAIRED TARGETS AND RELATIVE PREDICTION

Candidates and deployable features are fixed before branch measurements. In repetition r, candidate and incumbent branches start from the same restored environment and agent state. Each branch executes its first action and then follows π_b to the endpoint in Eq. (1). Its carry is updated from the executed actions and resulting observations. Branches share exogenous randomness where supported, without changing their marginal distributions. Repetitions use independent random streams.

Let $G_i^{(r)}$ be a complete discounted branch-return sample under the protocol and task scale defining J. It satisfies $\mathbb{E}[G_i^{(r)} \mid x, \mathcal{C}(x)] = J(x, u_i)$. We form

$$d_i^{(r)} = G_i^{(r)} - G_b^{(r)}, \qquad y_i = \frac{1}{R}\sum_{r=1}^{R} d_i^{(r)}, \qquad y_b = 0. \tag{2}$$

One fresh incumbent rollout is shared across candidates within each repetition. The labels are therefore dependent across candidates, but satisfy $\mathbb{E}[y_i \mid x, \mathcal{C}(x)] = \Delta_i(x)$. They measure the effect of changing the first action under a common continuation. The measurement audit assesses their variability.

Let $\phi_i = \phi(z, u_i)$ be the frozen descriptor of candidate u_i. It contains action and world-model features, but no measured branch returns. Fitting and deployment use the same imagined-continuation protocol. This protocol need not match the real baseline continuation used for labels. Member m uses a shared encoder $h_i^{(m)} = E_{\psi_m}(z, \phi_i)$ and ordered-pair features $\xi_{ij}^{(m)} = [h_i^{(m)}, h_j^{(m)}, h_i^{(m)} - h_j^{(m)}]$. It predicts

$$f_{m,i} = \frac{g_{\theta_m}(z, \xi_{ib}^{(m)}) - g_{\theta_m}(z, \xi_{bi}^{(m)})}{2}, \qquad f_{m,b} = 0. \tag{3}$$

The expected return difference changes sign when the two actions are exchanged under the same continuation. Antisymmetrization enforces this target symmetry. Measurement noise does not change the expected-value identity. For $M > 1$ independently initialized members (Lakshminarayanan et al., 2017), we compute $\mu_i = M^{-1}\sum_m f_{m,i}$ and $s_i^2 = (M-1)^{-1}\sum_m (f_{m,i} - \mu_i)^2$. The mean is the improvement score. Disagreement provides a candidate-dependent calibration scale.

3.3 CANDIDATE-SET DECISION-AWARE LEARNING

Comparison errors matter when they lead to costly action choices. We therefore train on complete candidate sets, including the incumbent. Let $y^\star = \max_{j\in\mathcal{I}} y_j$. We define the empirical selection cost as

$$c_i = y^\star - y_i + \lambda_{\mathrm{h}}[-y_i - \epsilon_{\mathrm{h}}]_+, \qquad i \in \mathcal{I}, \tag{4}$$

Here $\lambda_{\mathrm{h}}, \epsilon_{\mathrm{h}} \geq 0$. The first term is empirical regret. It penalizes forgone gains and negative measured improvements. The coefficient λ_{h} sets the extra penalty when $y_i < -\epsilon_{\mathrm{h}}$. The threshold ϵ_{h} uses normalized return units. Retaining the incumbent incurs only forgone improvement, since $c_b = y^\star$.

For member m, define $p_{m,i} = \exp(f_{m,i}/T)/\sum_{j\in\mathcal{I}} \exp(f_{m,j}/T)$ with temperature $T > 0$. We minimize the fitting-set average of

$$\mathcal{L}_m = \frac{1}{N}\sum_{i=1}^{N}(f_{m,i} - y_i)^2 + \lambda_{\mathrm{dec}} \sum_{i\in\mathcal{I}} p_{m,i}c_i + \lambda_{\mathrm{wd}}\|\Theta_m\|_2^2, \tag{5}$$

where $\Theta_m = (\psi_m, \theta_m)$ contains the trainable parameters and $\lambda_{\mathrm{dec}}, \lambda_{\mathrm{wd}} \geq 0$. Regression anchors scores to measured return differences. The decision term shifts softmax mass toward lower-cost choices. Since $\sum_i p_{m,i} = 1$, the shared term $y^\star$ does not affect the gradient. Candidates are coupled through the softmax normalization, which includes the incumbent.

The softmax is used only for training. Deployment uses the calibrated ensemble scores in Eq. equation 7. Decision-aware training can shift scores away from regression-only estimates. We therefore calibrate the final predictor. Minibatches retain complete candidate sets. Normalization, model selection, and hyperparameter tuning use fitting data and source-disjoint internal validation. All predictor parameters are frozen before calibration.

3.4 SIMULTANEOUS CALIBRATION OF MEASURED RETURN DIFFERENCES

We calibrate each task/checkpoint group separately. A source block contains its retained roots, candidates, and paired measurements. Our experiments retain one root per source episode, so each block is one complete candidate set. Fitting, calibration, and evaluation use disjoint source trajectories. Conditional on fitting, calibration and evaluation blocks are assumed exchangeable under the same source sampling, root selection, candidate generation, and R-repetition measurement protocol. The sample size n_{cal} counts source blocks.

Set $w_i(x) = s_i(x) + \varepsilon$ with fixed $\varepsilon > 0$. For each calibration block, compute

$$q_k = \max_{x\in\mathcal{B}_k} \max_{1\leq i\leq N} \frac{|\mu_i(x) - y_i(x)|}{w_i(x)}. \tag{6}$$

The maximum covers all comparisons in a block before action selection. Residuals include prediction error and finite-rollout variability. Calibration therefore targets the measured return differences y_i, not their expectations Δ_i.

Fix $\alpha \in (0, 1)$ before calibration and let $k_{\mathrm{cal}} = \lceil (n_{\mathrm{cal}} + 1)(1 - \alpha) \rceil$. Following split-conformal calibration (Angelopoulos & Bates, 2023), $\widehat{q}$ is the k_{cal}-th smallest element of $\{q_1, \ldots, q_{n_{\mathrm{cal}}}, +\infty\}$. Define

$$r_i(x) = \widehat{q}\, w_i(x), \qquad L_i(x) = \mu_i(x) - r_i(x), \qquad 1 \leq i \leq N, \qquad \mu_b = r_b = L_b = 0. \tag{7}$$

The ensemble, feature map, scale rule, and nominal level are fixed before calibration. An infinite quantile forces incumbent fallback. Deployment uses only candidate features, ensemble predictions, and the stored quantile.

3.5 REPLACEMENT RULE AND DECISION PROPERTIES

BRVL selects $\hat{i} \in \arg\max_{i\in\mathcal{I}} L_i$, with ties favoring the incumbent. It executes the highest-scoring alternative only when that score is positive. Otherwise, it retains the incumbent. Closed-loop deployment applies this rule at every decision interval.

We first analyze selection on fresh evaluation blocks. We then relate the baseline-relative target to closed-loop return. Under the preceding exchangeability assumption, the blockwise rank argument gives

$$\mathcal{E}_y = \bigcap_{x \in \mathcal{B}_{\text{test}}} \bigcap_{i \in \mathcal{I}} \{|\mu_i(x) - y_i(x)| \le r_i(x)\}, \qquad \Pr(\mathcal{E}_y) \ge 1 - \alpha. \tag{8}$$

The probability is marginal over calibration blocks and a fresh evaluation block, conditional on fitting. Selection uses no evaluation measurements. On $\mathcal{E}_y$, every retained root satisfies $y_{\hat{i}} \ge L_{\hat{i}} \ge 0$.

Marginal expected harmful loss. Fix one task/checkpoint group and sample a root from a fresh block under the evaluation protocol. Assume each fresh label has conditional mean Δ_i given the root, candidates, and all selection information. Let $0 \le B_{\text{h}} < \infty$ be a known bound fixed conditional on fitting. Assume $y_i \ge -B_{\text{h}}$ almost surely for all roots and candidates under this protocol. Conditional Jensen's inequality and Eq. (8) give

$$\mathbb{E}[[-\Delta_{\hat{i}}]_+] \le \mathbb{E}[[-y_{\hat{i}}]_+] \le \alpha B_{\text{h}}. \tag{9}$$

Expectations average over calibration, root sampling, candidate generation, and fresh measurements, conditional on fitting. The bound controls the average magnitude of negative expected improvements under this protocol.

Relation to closed-loop return. Fix the fitted selector and its calibration quantile. For a deterministic baseline with $u_b = \pi_b(x)$, $\Delta_i(x) = A^{\pi_b}(x, u_i)$. Let $\eta_\gamma(\pi)$ denote expected normalized discounted episode return. Both policies share the initial-state distribution, reward convention, task scale, discount, and finite endpoint. The state includes time and agent carry. Terminal value is zero. The performance-difference identity (Schulman et al., 2015a) gives

$$\eta_\gamma(\pi_{\text{BRVL}}) - \eta_\gamma(\pi_b) = \mathbb{E}_{\tau \sim \pi_{\text{BRVL}}} \left[\sum_{t=0}^{T_\tau - 1} \gamma^t \Delta_{\hat{i}_t}(x_t) \right]. \tag{10}$$

This identity sums baseline-relative advantages along BRVL trajectories. Offline coverage does not automatically extend to this state distribution.

4 Experiments

4.1 Experimental Setup

We evaluate frozen, task-specific DreamerV3 (Hafner et al., 2025) and TD-MPC2 (Hansen et al., 2024) agents with state-based observations. The benchmarks contain 39 DMControl tasks (Tassa et al., 2018), 50 Meta-World tasks (Yu et al., 2020), and five ManiSkill2 tasks (Gu et al., 2023). Each set contains the incumbent and $N = 9$ alternatives. DreamerV3 samples alternatives from its frozen actor. TD-MPC2 uses the first actions of the nine highest-scoring final elite sequences from one native planning call. The native controller returns the incumbent. Methods share candidate-generation rules and budgets. Frozen-set comparisons reuse identical saved candidates.

No replacement executes the incumbent. **Abs-Score** ranks all candidates using imagined rewards and a terminal value estimate, without branch supervision. **Abs-Reg+Gate** regresses absolute returns and subtracts the incumbent prediction within each ensemble member. **Rel-MSE+Gate** retains BRVL's targets, inputs, and ensemble architecture, but sets $\lambda_{\text{dec}} = 0$. **RVL-adapted+Gate** (Höftmann et al., 2026) adapts pairwise Bellman value learning to action-conditioned comparisons. **DAE-adapted+Gate** (Pan & Schölkopf, 2026) uses constrained off-policy return decomposition. These adaptations change the objective rather than isolate one BRVL component. Gated methods share replacement and calibration procedures, with predictor-specific quantiles.

Each learned method uses one shared selector per agent–benchmark pair. Each task/checkpoint group provides 60 fitting, 20 calibration, and 20 frozen-evaluation source episodes. Source splits are disjoint, with one root per episode. Internal validation uses fitting data only. All partitions use $R = 3$ paired repetitions per root. We evaluate three independent initializations of the complete selector ensemble. Each predictor, initialization, and task/checkpoint group is calibrated separately at $\alpha = 0.10$, giving $k_{\text{cal}} = 19$.

Evaluation uses new episodes of the fitted tasks. Frozen-set evaluation uses fresh $R = 3$ paired measurements, with π_b following each initial action. Closed-loop evaluation uses 20 paired episodes per task/checkpoint and selector initialization, disjoint from branch-collection sources. BRVL is reapplied at every decision, with $H_{\text{exec}} = 1$ and the native action repeat. We report episode return on DMControl and success rate on manipulation tasks. Tasks receive equal weight within each benchmark. We report point estimates and pointwise 95% percentile-bootstrap intervals for paired method differences using 10,000 resamples. Resampling preserves complete-selector runs across tasks and shared evaluation episodes and source blocks. Tasks, base checkpoints, and fitting and calibration data remain fixed. Across-initialization standard deviations and latency quantiles are reported separately.

4.2 Closed-Loop Performance and Decision Quality

Table 1 evaluates BRVL at two levels. Frozen-set evaluation isolates selection quality on common roots and candidates under baseline continuation. Closed-loop evaluation measures complete-episode performance under repeated selection. Methods share initial conditions in closed-loop evaluation.

Frozen-set metrics use $R = 3$ fresh paired repetitions. Evaluation labels are excluded from fitting, calibration, and action selection. They use the task scale in Eq. (1). We report

$$\begin{aligned} &\widehat{\Delta\bar{r}} = \mathrm{Avg}_x\, y_{\hat{i}}(x), && \widehat{D}_3 = \mathrm{Avg}_x[-y_{\hat{i}}(x)]_+, \\ &\widehat{H}_3 = \mathrm{Avg}_x\, \mathbf{1}\{y_{\hat{i}}(x) < -0.05\}. \end{aligned} \tag{11}$$

Net gain averages over all roots, including zero-gain incumbent decisions. It estimates candidate-set regret reduction relative to No replacement. Harm statistics use finite measurements, and $\widehat{H}_3$ is reported in percent. Measurement noise affects its variance. Harm statistics apply nonlinear functions to $y_{\hat{i}}$ and remain measurement-dependent. The separate $R_{\text{audit}} = 20$ audit evaluates decision quality with more repetitions. Frozen gain and closed-loop performance use different state distributions and outcome definitions.

Closed-loop results. BRVL has the highest mean closed-loop performance in all six settings (Table 1). Relative to No replacement, DMControl returns increase by 60.4 for DreamerV3 and 48.3 for TD-MPC2. Meta-World success increases by 21.9 and 1.9 percentage points, respectively. ManiSkill2 success increases by 34.3 and 7.3 percentage points. Abs-Score improves DreamerV3 on DMControl and ManiSkill2. It performs worse than No replacement on Meta-World and in all TD-MPC2 settings. Maximizing predicted return over the candidate set therefore does not consistently improve the native controller.

Frozen-set decision quality. BRVL outperforms Rel-MSE+Gate in mean closed-loop performance and frozen net gain in all six settings. The two methods share paired Monte Carlo labels, so this comparison controls for label type. BRVL also has lower measured harm on DMControl for both agents and on both manipulation benchmarks with DreamerV3. TD-MPC2 manipulation shows a gain–harm trade-off. Rel-MSE+Gate has lower harmful loss and severe-harm rates than BRVL, whereas DAE-adapted+Gate has higher frozen net gain. Frozen-set rankings need not match closed-loop rankings under repeated intervention.

4.3 Paired Analysis of Decision Changes

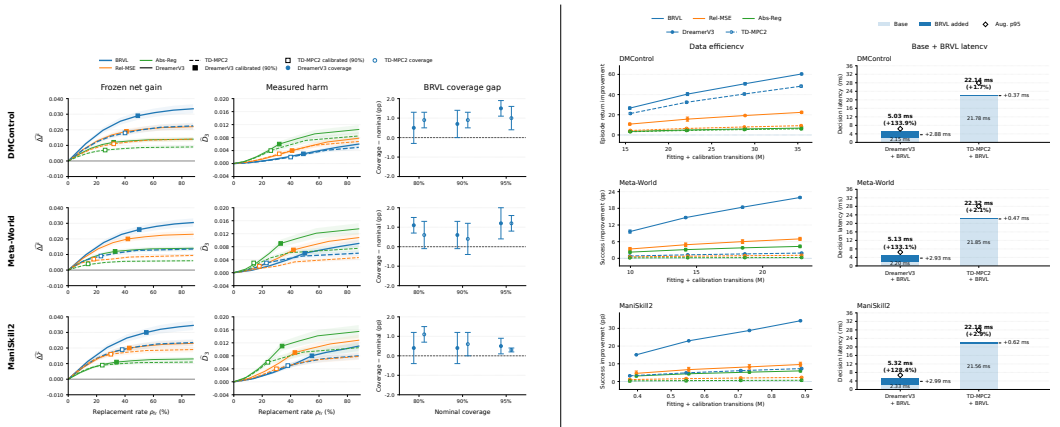
We compare BRVL (B) with Rel-MSE+Gate (A) on the same frozen roots and candidates. Both methods select actions before accessing evaluation measurements. They then share the same independent $R = 3$ paired measurements. Let $v_j(x) = y^{(3)}_{i_j}(x)$ for $j \in \{A, B\}$ and $\delta_A(x) = v_B(x) - v_A(x)$. We define favorable and unfavorable return contributions as

$$\widehat{C}^+_A = \mathrm{Avg}_x[\delta_A(x)]_+, \qquad \widehat{C}^-_A = \mathrm{Avg}_x[-\delta_A(x)]_+. \tag{12}$$

Their difference equals the frozen net-gain improvement: $\widehat{C}^+_A - \widehat{C}^-_A = \widehat{\Delta\bar{r}}_B - \widehat{\Delta\bar{r}}_A$. Unlike harmful-replacement loss, $\widehat{C}^-_A$ also includes forgone gains when both choices outperform the incumbent.

Table 1: Closed-loop performance and frozen-set decision quality. CL is episode return on DM-Control and success rate (%) on manipulation tasks. Frozen-set metrics use $R = 3$; $\widehat{H}_3$ is shown in percent. Bold marks the best estimate among replacement methods for each benchmark and agent.

	DreamerV3				TD-MPC2			
Method	CL↑	$\widehat{\Delta\bar{r}}\uparrow$	$\widehat{D}_3\downarrow$	$\widehat{H}_3\downarrow$	CL↑	$\widehat{\Delta\bar{r}}\uparrow$	$\widehat{D}_3\downarrow$	$\widehat{H}_3\downarrow$
DMControl (39 tasks; CL = episode return)								
No replacement	760.4	0.000	0.000	0.0	870.0	0.000	0.000	0.0
Abs-Score	766.9	0.010	0.014	7.2	868.7	−0.002	0.011	4.7
Abs-Reg+Gate	767.2	0.012	0.006	2.4	876.1	0.007	0.004	1.4
Rel-MSE+Gate	782.8	0.019	0.004	1.3	879.2	0.011	0.003	0.9
RVL-adapted+Gate	790.1	0.018	0.010	4.8	883.6	0.010	0.007	2.8
DAE-adapted+Gate	784.6	0.021	0.009	4.1	894.8	0.012	0.006	2.2
BRVL (ours)	**820.8**	**0.029**	**0.003**	**0.9**	**918.3**	**0.018**	**0.002**	**0.6**
Meta-World (50 tasks; CL = success rate, %)								
No replacement	41.9	0.000	0.000	0.0	94.0	0.000	0.000	0.0
Abs-Score	40.8	0.004	0.021	9.3	93.2	0.001	0.010	4.2
Abs-Reg+Gate	46.2	0.012	0.009	3.8	94.3	0.004	0.003	1.1
Rel-MSE+Gate	48.9	0.020	0.007	2.6	95.1	0.007	**0.002**	**0.6**
RVL-adapted+Gate	53.5	0.024	0.014	6.5	94.6	0.008	0.007	2.9
DAE-adapted+Gate	55.1	0.023	0.012	5.4	95.3	**0.010**	0.005	1.9
BRVL (ours)	**63.8**	**0.026**	**0.006**	**2.2**	**95.9**	0.009	0.003	0.8
ManiSkill2 (5 tasks; CL = success rate, %)								
No replacement	23.7	0.000	0.000	0.0	87.0	0.000	0.000	0.0
Abs-Score	28.7	0.005	0.027	11.4	86.7	−0.003	0.018	7.2
Abs-Reg+Gate	29.7	0.011	0.011	4.4	87.7	0.009	0.006	2.1
Rel-MSE+Gate	33.3	0.020	0.009	3.5	89.3	0.016	**0.004**	**1.5**
RVL-adapted+Gate	36.7	0.018	0.022	9.5	88.7	0.014	0.012	4.8
DAE-adapted+Gate	39.6	0.026	0.020	8.1	90.3	**0.021**	0.009	3.6
BRVL (ours)	**58.0**	**0.030**	**0.008**	**3.1**	**94.3**	0.019	0.005	1.8



(a) Calibration diagnostics (b) Efficiency

Figure 2: Calibration and efficiency diagnostics across three benchmarks. Bands and non-latency error bars show ±1 standard deviation across three selector initializations. Latency bars show means; diamonds show the 95th percentile.

Repair and new-harm rates count crossings of the incumbent's zero-improvement boundary:

$$\widehat{p}_{\text{repair}} = \text{Avg}_x\, \mathbf{1}\{v_A(x) < 0,\ v_B(x) \geq 0\},$$
$$\widehat{p}_{\text{new}} = \text{Avg}_x\, \mathbf{1}\{v_A(x) \geq 0,\ v_B(x) < 0\}. \tag{13}$$

All averages include every evaluation root and give equal weight to tasks. These diagnostics measure single interventions followed by π_b. Closed-loop performance is evaluated separately.

Table 2: Paired differences from Rel-MSE+Gate to BRVL. Brackets show pointwise 95% bootstrap intervals. δ_{CL} uses episode-return units on DMControl and percentage points on manipulation tasks. $\delta_{\mathrm{NG}} = \widehat{C}_A^+ - \widehat{C}_A^-$ is the frozen net-gain difference.

Benchmark	Agent	$\widehat{\delta}_{\mathrm{CL}}$	$\widehat{\delta}_{\mathrm{NG}}$	$\widehat{C}_A^+$	$\widehat{C}_A^-$
DMControl	DreamerV3	38.0 [30.0, 46.0]	0.010 [0.006, 0.014]	0.013	0.003
	TD-MPC2	39.1 [31.1, 47.1]	0.007 [0.003, 0.011]	0.009	0.002
Meta-World	DreamerV3	14.9 [10.9, 18.9]	0.006 [0.002, 0.010]	0.011	0.005
	TD-MPC2	0.8 [−0.6, 2.2]	0.002 [−0.001, 0.005]	0.006	0.004
ManiSkill2	DreamerV3	24.7 [18.7, 30.7]	0.010 [0.004, 0.016]	0.017	0.007
	TD-MPC2	5.0 [1.0, 9.0]	0.003 [−0.001, 0.007]	0.008	0.005

Table 3: Ablations on ten DMControl tasks. Return is the mean closed-loop episode return. Frozen-set metrics use $R = 3$, with $\widehat{H}_3$ in percent. Bold marks the best point estimate among replacement methods for each agent.

	DreamerV3				TD-MPC2			
Method	Return↑	$\widehat{\Delta\bar{r}}\uparrow$	$\widehat{D}_3\downarrow$	$\widehat{H}_3\downarrow$	Return↑	$\widehat{\Delta\bar{r}}\uparrow$	$\widehat{D}_3\downarrow$	$\widehat{H}_3\downarrow$
No replacement	690.0	0.000	0.000	0.0	860.0	0.000	0.000	0.0
Rel-MLP+Gate	735.3	0.017	0.006	2.2	890.6	0.010	0.004	1.3
Rel-MSE+Gate	712.9	0.019	0.005	1.5	867.8	0.012	0.003	1.0
BRVL-Mean	742.5	0.022	0.006	1.8	899.4	0.014	0.004	1.2
Calibrated-LCB+Forced	720.7	0.020	0.011	3.7	878.2	0.013	0.008	2.5
BRVL (ours)	**749.0**	**0.028**	**0.003**	**0.8**	**906.0**	**0.018**	**0.002**	**0.7**

Decision-change patterns. BRVL has higher point estimates of closed-loop performance and frozen net gain than Rel-MSE+Gate in all six settings (Table 2). Favorable contributions outweigh unfavorable ones, yielding frozen net-gain improvements of 0.002–0.010. On TD-MPC2 manipulation tasks, unfavorable changes offset a larger share, leaving improvements of 0.002–0.003. BRVL thus improves the balance of return changes without eliminating unfavorable decisions.

4.4 Ablations and Calibration

We evaluate ablations on ten prespecified DMControl tasks for both agents. Shared predictors still use the full 39-task fitting data. **Rel-MLP+Gate** removes enforced antisymmetry but retains the pair features and full objective. **Rel-MSE+Gate** removes only the decision-aware loss. **BRVL-Mean** reuses the BRVL ensemble and removes the calibrated margin, but retains the zero-valued incumbent. **Calibrated-LCB+Forced** retains calibrated scores but excludes the incumbent. Retrained variants are calibrated separately. Closed-loop evaluations share initial conditions. Frozen-set comparisons share roots and candidates.

Component effects. BRVL has the best point estimate on all four metrics in this ten-task subset. Removing the decision-aware loss lowers mean return by 36.1 for DreamerV3 and 38.2 for TD-MPC2. Frozen net gain falls by 0.009 and 0.006, respectively. Removing antisymmetry lowers mean return by 13.7 and 15.4 points. Removing the calibrated margin doubles measured harmful loss. Removing incumbent fallback raises this loss to 0.011 and 0.008.

Calibration diagnostics. Figure 2(a) reports frozen net gain and measured harm against replacement rate $\widehat{\rho}_{\mathrm{fr}}$. Thresholds use fitting-only validation. For scores $\mu_i - \lambda w_i$, calibration sets $\lambda = \widehat{q}$. With $n_{\mathrm{cal}} = 20$ and $\alpha = 0.10$, $\widehat{q}$ is the second-largest finite calibration score. Conservatism depends on $r_i = \widehat{q} w_i$, not $\widehat{q}$ alone. The curves compare decision quality across replacement rates. The final column reports simultaneous measured-coverage gaps on fresh $R = 3$ candidate sets. These diagnostics concern frozen roots, not closed-loop visits.

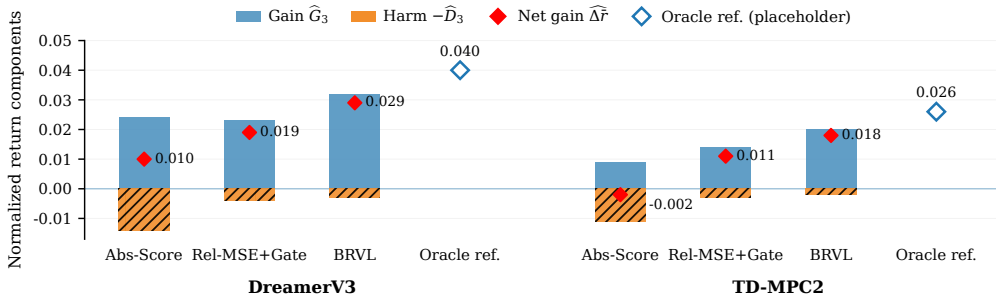


Figure 3: Gain–harm decomposition on DMControl using the main frozen-set evaluation with $R = 3$ paired repetitions. Bars show $\widehat{G}_3$ and $-\widehat{D}_3$; diamonds show frozen net gain.

4.5 Data and Computational Efficiency

Offline supervision includes source episodes and branch rollouts for fitting and calibration. With $N = 9$ and $R = 3$, each root requires 30 branch trajectories before merging exact duplicate actions. The 60 fitting and 20 calibration roots therefore require 2,400 trajectories per task/checkpoint group before reuse. Evaluation and audit costs are counted separately.

Figure 2(b) compares complete native and augmented decision paths. The augmented path includes candidate processing, imagined features, ensemble prediction, and selection. Reused native work is counted once. Added latency is the difference between path means. Relative overhead divides this difference by the native mean. Calibration is fitted offline and contributes one stored quantile per task/checkpoint group. Root acquisition and branch rollouts are separate offline costs.

4.6 Selected-Action Utility and Reference Gap

We audit prespecified roots from ten DMControl tasks. With predictors and selected actions fixed, we collect $R_{\text{audit}} = 20$ new paired repetitions per candidate. Each branch executes one action and then follows π_b. These repetitions are independent of the primary $R = 3$ measurements and are not pooled with them. We decompose net gain using $y_{\hat{i}}^{(20)}$:

$$\begin{aligned} \widehat{G}_{20} &= \mathrm{Avg}_x[y_{\hat{i}}^{(20)}(x)]_+, \qquad \widehat{D}_{20} = \mathrm{Avg}_x[-y_{\hat{i}}^{(20)}(x)]_+, \\ \widehat{\Delta\bar{r}}_{20} &= \widehat{G}_{20} - \widehat{D}_{20}. \end{aligned} \tag{14}$$

All metrics use common root and task weights. The cross-fitted reference selects the candidate with the highest measured improvement from ten repetitions, including the incumbent. It evaluates that choice on the other ten. We swap the halves and average. This reference is not a certified upper bound, and its gap to BRVL is not exact regret.

Figure 3 reports BRVL net gains of 0.028 and 0.018 for DreamerV3 and TD-MPC2, respectively. Positive contributions of about 0.031 and 0.020 outweigh harmful losses of 0.003 and 0.002. Harm offsets about 10% of positive contributions for either agent. Reference gaps of 0.012 and 0.008 suggest scope for better selection within the same candidate sets, but do not measure the generator's performance ceiling or closed-loop recoverability.

5 Conclusion and Limits

BRVL combines paired relative-return learning, decision-aware training, and blockwise calibration for post-training selection. It improves mean closed-loop performance and frozen net gain over matched-supervision regression baselines in all six settings. BRVL requires restorable simulators and extra branch rollouts. Gains depend on available candidates; offline calibration does not certify closed-loop improvement. Evaluation covers new episodes of fitted tasks. Future work targets deployment-state calibration, richer candidates, cheaper branch collection, and unseen-task transfer.

REFERENCES

Anastasios Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. Conformal risk control. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/f3549ef9b5ff520a7e41ff3cc306ab2b-Abstract-Conference.html.

Anastasios N. Angelopoulos and Stephen Bates. Conformal prediction: A gentle introduction. *Foundations and Trends in Machine Learning*, 16(4):494–591, 2023. doi: 10.1561/2200000101. URL https://doi.org/10.1561/2200000101.

Kavosh Asadi, Dipendra Misra, Seungchan Kim, and Michael L. Littman. Combating the compounding-error problem with a multi-step model. *arXiv preprint arXiv:1905.13320*, 2019.

Dimitri P. Bertsekas. Differential training of rollout policies. In *Proceedings of the 35th Allerton Conference on Communication, Control, and Computing*, Allerton Park, Illinois, October 1997. URL https://faculty.engineering.asu.edu/sites/g/files/litvpz5506/files/2026-06/Diftrain.pdf.

Jacob Buckman, Danijar Hafner, George Tucker, Eugene Brevdo, and Honglak Lee. Sample-efficient reinforcement learning with stochastic ensemble value expansion. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Wei-Di Chang, Mikael Henaff, Brandon Amos, Gregory Dudek, and Scott Fujimoto. The surprising difficulty of search in model-based reinforcement learning. In *International Conference on Machine Learning (ICML)*, 2026. URL https://icml.cc/virtual/2026/poster/62452.

Kurtland Chua, Roberto Calandra, Rowan McAllister, and Sergey Levine. Deep reinforcement learning in a handful of trials using probabilistic dynamics models. In *Advances in Neural Information Processing Systems*, 2018.

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

Amir-massoud Farahmand. Action-gap phenomenon in reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 24, pp. 172–180, 2011.

Vladimir Feinberg, Alvin Wan, Ion Stoica, Michael I. Jordan, Joseph E. Gonzalez, and Sergey Levine. Model-based value estimation for efficient model-free reinforcement learning. *arXiv preprint arXiv:1803.00101*, 2018.

Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://papers.nips.cc/paper_files/paper/2017/hash/4a8423d5e91fda00bb7e46540e2b0cf1-Abstract.html.

Jiayuan Gu, Fanbo Xiang, Xuanlin Li, Zhan Ling, Xiqiang Liu, Tongzhou Mu, Yihe Tang, Stone Tao, Xinyue Wei, Yunchao Yao, Xiaodi Yuan, Pengwei Xie, Zhiao Huang, Rui Chen, and Hao Su. ManiSkill2: A unified benchmark for generalizable manipulation skills. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2302.04659.

David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. In *Advances in Neural Information Processing Systems 31*, pp. 2455–2467. Curran Associates, Inc., 2018. URL https://papers.nips.cc/paper/7512-recurrent-world-models-facilitate-policy-evolution.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. In *International Conference on Learning Representations*, 2020.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, 640:647–653, 2025. doi: 10.1038/s41586-025-08744-2.

Nicklas Hansen, Hao Su, and Xiaolong Wang. Td-mpc2: Scalable, robust world models for continuous control. In *International Conference on Learning Representations*, 2024.

Marc Höftmann, Jan Robine, and Stefan Harmeling. Relative value learning. In *International Conference on Learning Representations*, 2026.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/9ef2ed4b7fd2c810847ffa5fa85bce38-Abstract.html.

Romain Laroche, Paul Trichelair, and Remi Tachet Des Combes. Safe policy improvement with baseline bootstrapping. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3652–3661. PMLR, 2019.

Alessandro Lazaric, Mohammad Ghavamzadeh, and Rémi Munos. Analysis of classification-based policy iteration algorithms. *Journal of Machine Learning Research*, 2016.

Khang Nguyen, An T. Le, Jan Peters, and Minh Nhat Vu. DoublyAware: Dual planning and policy awareness for temporal difference learning in humanoid locomotion. *IEEE Robotics and Automation Letters*, 11(2):2162–2169, 2026. doi: 10.1109/LRA.2025.3648611. URL https://doi.org/10.1109/LRA.2025.3648611.

Daniel Palenicek, Michael Lutter, João Carvalho, Daniel Dennert, Faran Ahmad, and Jan Peters. Diminishing return of value expansion methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2026. doi: 10.1109/TPAMI.2026.3713917. URL https://doi.org/10.1109/TPAMI.2026.3713917. Early Access.

Hsiao-Ru Pan and Bernhard Schölkopf. Skill or luck? return decomposition via advantage functions. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=ZFMiHfZwIf.

Hsiao-Ru Pan and Bernhard Schölkopf. Direct advantage estimation for scalable and sample-efficient deep reinforcement learning. *Reinforcement Learning Journal*, 7, 2026. URL https://rlj.cs.umass.edu/2026/papers/Paper11.html.

Hsiao-Ru Pan, Nico Gürtler, Alexander Neitz, and Bernhard Schölkopf. Direct advantage estimation. In *Advances in Neural Information Processing Systems*, volume 35, 2022. doi: 10.52202/068431-0862. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/4d893f766ab60e5337659b9e71883af4-Abstract-Conference.html.

Marek Petrik, Yinlam Chow, and Mohammad Ghavamzadeh. Safe policy improvement by minimizing robust baseline regret. In *Advances in Neural Information Processing Systems*, volume 29, pp. 2298–2306, 2016.

Drew Prinster, Clara Fannjiang, Ji Won Park, Kyunghyun Cho, Anqi Liu, Suchi Saria, and Samuel Stanton. Conformal policy control. In *Forty-third International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2603.02196.

Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, et al. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588:604–609, 2020.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 1889–1897. PMLR, 2015a.

John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015b.

Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, Timothy Lillicrap, and Martin Riedmiller. DeepMind control suite. *arXiv preprint arXiv:1801.00690*, 2018. URL https://arxiv.org/abs/1801.00690.

Yuhang Wang, Hanwei Guo, Sizhe Wang, Long Qian, and Xuguang Lan. Bootstrapped model predictive control. In *The Thirteenth International Conference on Learning Representations*. OpenReview.net, 2025. URL https://openreview.net/forum?id=i7jAYFYDcM.

Ziyu Wang, Tom Schaul, Matteo Hessel, Hado van Hasselt, Marc Lanctot, and Nando de Freitas. Dueling network architectures for deep reinforcement learning. In *International Conference on Machine Learning*, pp. 1995–2003, 2016.

Shoki Yamao, Ken Kobayashi, Ryo Matsui, Shota Nagai, Naoki Nishimura, and Kazuhide Nakata. Robust decision-focused learning via worst-case regret minimization. In Emilija Perković and Daniel Malinsky (eds.), *Proceedings of the 42nd Conference on Uncertainty in Artificial Intelligence*, volume 337 of *Proceedings of Machine Learning Research*, pp. 7701–7736. PMLR, 2026.

Christopher Yeh, Nicolas Christianson, Adam Wierman, and Yisong Yue. Conformal risk training: End-to-end optimization of conformal risk control. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-2355. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/6559542f75b4452ebaaf82094c7defb7-Abstract-Conference.html.

Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-World: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Proceedings of the Conference on Robot Learning*, volume 100 of *Proceedings of Machine Learning Research*, pp. 1094–1100. PMLR, 2020. URL https://proceedings.mlr.press/v100/yu20a.html.

Guojian Zhan, Likun Wang, Xiangteng Zhang, Jiaxin Gao, Masayoshi Tomizuka, and Shengbo Eben Li. Bootstrap off-policy with world model. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://arxiv.org/abs/2511.00423.

AI USE STATEMENT

We did not use generative AI tools for any task requiring disclosure: no synthetic datasets, theoretical models, mathematical claims, proof ingredients or proof writing, hypotheses, research methodology or experimental design, method implementation, translation, dataset cleaning and reformatting, or result interpretation were produced with such tools, and these tasks are not applicable to this work. Generative AI was used only for the recommended-disclosure tasks of (i) copy-editing the manuscript for readability, (ii) formatting references, and (iii) identifying and checking relevant literature. All AI-assisted output was manually verified by the authors: every bibliographic entry was checked against DBLP, arXiv, PMLR, and publisher records, entries with incorrect titles, volumes, or page numbers were corrected, entries that could not be verified against any bibliographic database were removed, and AI-suggested wording was reviewed line by line. No experimental results, figures, tables, or mathematical statements were generated by generative AI tools, and the authors take responsibility for the final content of this work, including any text, claims, or artifacts produced with AI assistance.

A ADDITIONAL EXPERIMENTAL DETAILS

A.1 BENCHMARKS AND EXPERIMENTAL PROTOCOL

We use state-based observations on 39 DMControl tasks, 50 Meta-World tasks, and five ManiSkill2 tasks. The ManiSkill2 tasks are LiftCube, PickCube, StackCube, PickSingleYCB, and TurnFaucet. A group consists of one task and one frozen base-agent checkpoint. Methods in each group share preprocessing, native evaluation settings, candidate budgets, and evaluation sources. Base-agent parameters and native planner settings remain fixed. Table 5 lists the selector and evaluation settings. Unavailable groups are excluded from all paired comparisons, not assigned zero scores.

Table 4: Fixed experimental settings.

Item	Setting
Candidate positions	Incumbent plus $N = 9$ alternatives
Execution	$H_{\text{exec}} = 1$ with native action repeat
Paired branch measurements	$R = 3$ per root
Source episodes per group	100; one eligible root per episode
Fitting / calibration / evaluation	60 / 20 / 20 source episodes
Internal validation	Source-disjoint subset of the 60 fitting episodes
Closed-loop evaluation	20 paired episodes per group and selector initialization
Complete-selector initializations	3; ensemble size M specified separately
Primary miscoverage	$\alpha = 0.10$
Diagnostic coverage levels	0.80, 0.90, 0.95
Imagined scoring horizon	DreamerV3: 15; TD-MPC2: 3
Native planning horizon	Unchanged from the frozen controller

Each agent–benchmark pair uses a shared selector. We evaluate three initializations of the complete selector ensemble. Each ensemble has $M = 5$ members. Calibration is separate for each predictor, initialization, and task/checkpoint group. Evaluation uses new episodes of the fitted tasks, not held-out tasks.

A.2 Decision-Loss Gradients and Measurement Noise

Fix the measured labels. Define $\widetilde{c}_i = -y_i + \lambda_{\text{h}}[-y_i - \epsilon_{\text{h}}]_+$ and $\overline{c}_m = \sum_i p_{m,i}\widetilde{c}_i$. The softmax includes the incumbent and sums to one, so

$$\mathcal{L}_{\text{dec},m} = \sum_i p_{m,i}c_i = y^\star + \sum_i p_{m,i}\widetilde{c}_i, \qquad \frac{\partial \mathcal{L}_{\text{dec},m}}{\partial f_{m,k}} = \frac{p_{m,k}}{T}(\widetilde{c}_k - \overline{c}_m), \quad k = 1, \dots, N. \tag{15}$$

The shared maximum $y^\star$ has no effect on the gradient. Cross-fitting or bias-correcting this term alone would therefore leave training unchanged. Noise in individual labels still affects fitting, especially through the nonlinear harm penalty. Conditional on a fixed root and candidate set, unbiased labels and convexity give

$$\mathbb{E}[[-y_i - \epsilon_{\text{h}}]_+ \mid x, \mathcal{C}(x)] \geq [-\Delta_i - \epsilon_{\text{h}}]_+. \tag{16}$$

The decision term therefore optimizes a finite-measurement loss, not the exact expected-return loss. This sensitivity comes from individual labels rather than the shared maximum.

A.3 Candidates, Scoring, and Branch Execution

Candidates. DreamerV3 retains its deterministic native evaluation action u_b and samples nine alternatives from the frozen actor at the same root. TD-MPC2 retains the output of one native planning call. Its alternatives are the first actions of the nine highest-scoring final elite sequences. Ties use a fixed order, and planner budgets remain unchanged. An insufficient elite pool is flagged; no extra planning call is made. Candidates and deployable features are fixed before branch measurement.

All ten candidate positions are retained. Actions may share a measured branch only when both their effective actions and carry semantics are identical. An exact incumbent duplicate has label zero. Near-duplicate tolerances are used only to report diversity; they do not change labels. Replacement rates count effective action changes, not candidate-index changes.

Imagined scores. For candidate u_i, the world-model score is

$$S_i = \widehat{\mathbb{E}}\left[\sum_{t=0}^{H_{\text{plan}}-1} \widehat{\Gamma}_{i,t}\widehat{r}_{i,t} + \widehat{\Gamma}_{i,H_{\text{plan}}}\widehat{V}_{\text{term}}(\widehat{z}_{i,H_{\text{plan}}})\right], \tag{17}$$

Here $\widehat{\Gamma}_{i,t}$ is the cumulative scoring discount. The first action is u_i. Later imagined actions follow the frozen actor or the TD-MPC2 policy prior. All methods share the rollout count, reward and

value decoding, critic aggregation, and discount convention. The scoring horizon H_{plan} does not change the native planning horizon. Real branches resume the actual controller, including MPC for TD-MPC2. Score error may therefore include continuation mismatch.

Restoration and randomness. Each root stores the simulator, task, wrapper, timeout, random-generator, and history-dependent agent state. One native controller call produces u_b and a common post-decision carry. Additional candidate processing leaves this carry and the native random stream unchanged. Each branch restores the root and carry, then executes one candidate action with the native action repeat. It updates action history with the executed action and follows π_b until termination or timeout. TD-MPC2 does not use an alternative elite's remaining sequence as its warm start. Restoration-only fields are excluded from selector inputs.

Candidate and incumbent branches share exogenous randomness within each repetition where supported. This coupling preserves each branch's marginal distribution. Repetitions use independent future streams for stochastic components. Simulator, controller, candidate-generation, and model-scoring streams are separate. We compare the incumbent-only wrapper with native actions, rewards, termination, and carry under matched randomness. Exact deterministic replays add no measurement information.

A.4 Offline Data and Return Targets

We sample one eligible nonterminal root uniformly from each independent baseline source episode. This gives an episode-uniform root distribution. Sources are split before fitting, and validation uses only the fitting partition. All candidates, repetitions, transitions, and cached features from one source remain in the same partition. Preprocessing and predictor selection do not use calibration or evaluation outcomes.

Let $G^{(r)}_{i,\text{raw}}$ be the complete discounted branch return under the registered decision-time reward and termination convention. Real labels use the defined task endpoint without a learned tail bootstrap. For task k, define

$$\begin{aligned} \kappa_k &= \max\left\{Q_{0.95}(\mathcal{G}_k^{\text{opt}}) - Q_{0.05}(\mathcal{G}_k^{\text{opt}}), 1\right\}, \\ d_i^{(r)} &= \frac{G^{(r)}_{i,\text{raw}} - G^{(r)}_{b,\text{raw}}}{\kappa_k}, \qquad y_i^{(R)} = \frac{1}{R}\sum_{r=1}^{R} d_i^{(r)}, \qquad y_b^{(R)} = 0. \end{aligned} \tag{18}$$

The set $\mathcal{G}_k^{\text{opt}}$ pools raw branch-mean returns from optimization sources across both agent families. It excludes internal validation. The scale is shared across methods and splits, and normalization is applied once. Absolute-regression targets are $\bar{G}_i = (R\kappa_k)^{-1}\sum_r G^{(r)}_{i,\text{raw}}$.

The incumbent branch is shared within each repetition. Per-repetition candidate vectors are retained to preserve their dependence; transition-level observations, actions, rewards, discounts, endpoints, and carry information are also cached for adapted baselines. Pilot sources used during development are excluded from confirmatory evaluation.

A.5 Selector Fitting and Baselines

Architecture and fitting. BRVL uses the shared encoder and antisymmetric head in Eq. equation 3. Task/checkpoint-specific projections map z and candidate descriptors to 128-dimensional features. Table 5 lists the network and optimization settings. Minibatches retain complete candidate sets and weight task/checkpoint groups equally. Training uses Eq. equation 5, with an explicit squared-parameter penalty and zero optimizer weight decay. Ensemble disagreement is the sample standard deviation across members. Optimization and source-disjoint validation determine the final predictor, which is frozen before calibration.

Core controls. **No replacement** executes the saved native incumbent. **Abs-Score** maximizes S_i over the incumbent and alternatives without branch-supervised fitting or calibration. Ties favor the incumbent. If native selection uses exactly the same candidates, score, and tie rule, Abs-Score coincides with No replacement.

For **Abs-Reg+Gate**, member m minimizes absolute-return MSE plus the declared regularization and forms differences within the member:

$$\mathcal{L}_{\mathrm{abs},m} = \mathrm{Avg}_{x\in\mathcal{D}_{\mathrm{opt}}} \frac{1}{N+1}\sum_{i\in\mathcal{I}} (\widehat{J}_m(z,u_i) - \bar{G}_i)^2,$$
$$f_{m,i} = \widehat{J}_m(z,u_i) - \widehat{J}_m(z,u_b). \tag{19}$$

The incumbent contributes once per candidate set. **Rel-MSE+Gate** retains BRVL's inputs, relative targets, ensemble, and replacement rule, but sets $\lambda_{\mathrm{dec}} = 0$. Both controls use the same supervision and specified capacity and optimization budgets. Each refitted predictor receives its own calibration quantile.

RVL-adapted+Gate. RVL-adapted+Gate (Höftmann et al., 2026) uses an action-conditioned relative critic $F_m(\eta_i, \eta_j)$, where $\eta = (z, u)$. Its paired Bellman targets and loss are

$$t_{ij} = \widetilde{r}_i - \widetilde{r}_j + \gamma_k F_{\bar{m}}(\eta_i^+, \eta_j^+),$$
$$\mathcal{L}_{\mathrm{RVL},m} = \mathrm{Avg}_{(i,j)}[F_m(\eta_i,\eta_j) - t_{ij}]^2. \tag{20}$$

Rewards are normalized once. The target critic is $F_{\bar{m}}$, and successor actions follow the frozen controller. Terminal branches use zero-value absorbing states. If only one branch terminates, the live branch retains its bootstrap. Deployment uses $f_{m,i} = F_m((z,u_i),(z,u_b))$.

DAE-adapted+Gate. DAE-adapted+Gate (Pan & Schölkopf, 2024; 2026) uses constrained return decomposition. Let $A_{m,t} = A_m(z_t, a_t)$ and $B_{m,t} = B_m(z_t, a_t, r_t, z_{t+1})$. Its objective and constraints are

$$\mathcal{L}_{\mathrm{DAE},m} = \mathrm{Avg}_\tau \left[\sum_{t=0}^{T_\tau - 1} \gamma_k^t(\widetilde{r}_t - A_{m,t} - B_{m,t}) - V_m(z_0)\right]^2,$$
$$\mathbb{E}_{a\sim\pi_b(\cdot|z)} A_m(z,a) = 0, \qquad \mathbb{E}_{(r,z')\sim P(\cdot|z,a)} B_m(z,a,r,z') = 0. \tag{21}$$

Here z includes the specified history information, and terminal value is zero at the finite endpoint. Deployment uses $f_{m,i} = A_m(z,u_i) - A_m(z,u_b)$. Action centering uses the actual controller, not uniform candidates or a substituted policy prior. RVL and DAE are adapted references, not single-component controls. Their transition supervision, auxiliary parameters, and compute are counted separately.

A.6 Candidate Opportunity and Reference Gaps

At a fixed root, define candidate-set opportunity and selection regret as

$$\mathcal{H}_{\mathcal{C}}(x) = \max_{i\in\mathcal{I}} \Delta_i(x), \qquad \mathcal{R}_{\mathcal{C}}(x) = \mathcal{H}_{\mathcal{C}}(x) - \Delta_{\hat{i}}(x), \qquad \Delta_{\hat{i}}(x) = \mathcal{H}_{\mathcal{C}}(x) - \mathcal{R}_{\mathcal{C}}(x). \tag{22}$$

The incumbent ensures $\mathcal{H}_{\mathcal{C}} \geq 0$. An alternative may improve on the incumbent even when both have low absolute returns. The selector cannot create an improving action that is absent from its candidate set. The cross-fitted reference evaluates a finite-measurement selection rule within that set. Its gap to BRVL is not exact candidate-set headroom or a closed-loop recovery bound. A one-action intervention followed by π_b may also miss improvements that require several coordinated deviations.

A.7 Calibration and Replacement

With one root per source, each calibration block is one complete candidate set. For every fitted predictor, initialization, and task/checkpoint group, use

$$q_j = \max_{1\leq i\leq N} \frac{|\mu_i(x_j) - y_i^{(3)}(x_j)|}{s_i(x_j) + \varepsilon}, \qquad \ell = \lceil (n_{\mathrm{cal}} + 1)(1-\alpha) \rceil, \tag{23}$$

where $\varepsilon > 0$ is fixed before calibration. The quantile $\widehat{q}$ is the ℓ-th order statistic of $\{q_1, \ldots, q_{n_{\mathrm{cal}}}, +\infty\}$, without interpolation. Set $w_i = s_i + \varepsilon$, $L_i = \mu_i - \widehat{q} w_i$ for alternatives, and

$L_b = 0$. Execute the largest positive alternative score; otherwise retain the incumbent. Alternative ties follow fixed candidate order.

For $n_{\mathrm{cal}} = 20$, coverage levels 0.80, 0.90, and 0.95 use indices 17, 19, and 20. The 0.99 boundary check gives an infinite quantile and fallback. Coverage is evaluated as

$$\widehat{\mathrm{Coverage}} = \mathrm{Avg}_x \, \mathbf{1}\Big\{\forall i = 1, \ldots, N : \; |\mu_i(x) - y_i^{(3)}(x)| \le \widehat{q} w_i(x)\Big\}. \tag{24}$$

Coverage gaps are $100[\widehat{\mathrm{Coverage}} - (1-\alpha)]$ percentage points. This is simultaneous coverage of fresh three-repetition measurements under within-group exchangeability, not expected-return coverage or a closed-loop guarantee.

Threshold diagnostics. Mean+Threshold replaces the incumbent when the largest alternative mean exceeds $\tau \ge 0$. Raw-LCB+Fallback uses $\mu_i - \lambda w_i$ with the positive-score gate. Threshold grids and deployment settings are selected using fitting-only validation. Selection maximizes net gain, includes an always-incumbent option, and breaks ties toward less replacement. Test sweeps report achieved replacement rates without test-time tuning or retrospective rate matching. Each calibrated point uses its predictor's $\widehat{q}$. Setting $\lambda = \widehat{q}$ gives the same scores. Calibration is therefore evaluated through operating-point selection and coverage, not a different ranking rule.

A.8 Relating Frozen Gains to Closed-Loop Performance

Fix the fitted selector and its calibration quantile. Let V^{π_b} use the same reward scale, discount, and endpoint as Δ_i. The augmented state includes time and agent carry. Both policies use the same transition and carry-update rules. For a deterministic baseline,

$$\Delta_{\hat{i}_t}(x_t) = \mathbb{E}\big[r_t + \gamma V^{\pi_b}(x_{t+1}) - V^{\pi_b}(x_t) \mid x_t, u_{\hat{i}_t}\big].$$

Taking expectations along BRVL trajectories and summing gives

$$\begin{aligned}
&\mathbb{E}_{\tau \sim \pi_{\mathrm{BRVL}}}\left[\sum_{t=0}^{T_\tau - 1} \gamma^t \Delta_{\hat{i}_t}(x_t)\right] \\
&\quad = \mathbb{E}_{\tau \sim \pi_{\mathrm{BRVL}}}\left[\sum_{t=0}^{T_\tau - 1} \gamma^t r_t - V^{\pi_b}(x_0) + \gamma^{T_\tau} V^{\pi_b}(x_{T_\tau})\right].
\end{aligned}$$

The intermediate value terms cancel. Terminal value is zero. Both policies share the same initial-state distribution, so $\mathbb{E}[V^{\pi_b}(x_0)] = \eta_\gamma(\pi_b)$. This proves Eq. equation 10.

The identity evaluates advantages on BRVL trajectories. Frozen net gain instead averages single-intervention measurements over offline roots. It already includes zero contributions from fallback. Multiplying it by replacement rate or episode length does not generally recover episode improvement. Manipulation success also differs from normalized return.

The identity does not transfer offline calibration to deployment states. Repeated intervention can change the visited-state distribution. A low average replacement rate does not by itself ensure a small distribution shift. One early change may affect many later states. A high replacement rate also does not by itself establish calibration failure.

A.9 Evaluation and Diagnostic Estimates

Frozen-set metrics. Selectors choose actions before accessing independent evaluation returns. Average paired repetitions within each root before taking positive parts or thresholding:

$$\begin{aligned}
&\widehat{\Delta\bar{r}} = \mathrm{Avg}_x \, y_{\hat{i}}^{(3)}(x), && \widehat{G}_3 = \mathrm{Avg}_x [y_{\hat{i}}^{(3)}(x)]_+, \\
&\widehat{D}_3 = \mathrm{Avg}_x [-y_{\hat{i}}^{(3)}(x)]_+, && \widehat{H}_3 = \mathrm{Avg}_x \, \mathbf{1}\{y_{\hat{i}}^{(3)}(x) < -0.05\}.
\end{aligned} \tag{25}$$

Net gain estimates candidate-set regret reduction without identifying the best candidate. Tables report $100\widehat{H}_3$ in percent. Harm means return loss, not physical danger. The severe-harm threshold uses normalized return units and differs from the training tolerance ϵ_{h}. Frozen replacement $\widehat{\rho}_{\mathrm{fr}}$ counts effective action changes. With the same weights, conditional harmful loss is $\widehat{D}_3 / \widehat{\rho}_{\mathrm{fr}}$ when $\widehat{\rho}_{\mathrm{fr}} > 0$. Unrounded summaries satisfy $\widehat{\Delta\bar{r}} = \widehat{G}_3 - \widehat{D}_3$ and $\widehat{D}_3 \ge 0.05 \widehat{H}_3$. Audit utility plots apply the same decomposition to the separate $R_{\mathrm{audit}} = 20$ measurements.

Closed-loop endpoints and aggregation. Closed-loop evaluations use the paired initial conditions in Table 4, disjoint from branch collection, and reapply the selector after every decision. DMControl reports unnormalized episode return; manipulation reports binary episode success in percent, with changes in percentage points. Manipulation returns are secondary. Frozen single-intervention gain and complete-episode improvement are separate endpoints.

We first average roots or episodes within each task/checkpoint and selector initialization. We then average initializations and checkpoints within each task, followed by an equal-weight average across tasks. Benchmarks retain their own outcome units. Closed-loop replacement is the fraction of eligible decisions with an effective action change within each group and initialization. It uses the same outer weights. Reused outcomes from training-free baselines do not count as new independent observations.

Uncertainty. We use $10{,}000$ paired bootstrap replicates for pointwise 95% intervals and method contrasts. The evaluated task/checkpoint set remains fixed. Complete-selector identities are resampled jointly across tasks that share a predictor. Source-root or episode blocks are resampled within groups. Compared methods share resampling indices, as do evaluations that reuse sources. Candidate and repetition vectors remain intact. Intervals condition on the fitting and calibration data and evaluated checkpoints; they do not include base-agent retraining. Three selector initializations give limited resolution for between-fit variation. Captions distinguish confidence intervals, across-initialization standard deviations, and latency quantiles.

Measurement audit and reference. The reliability audit uses five prespecified roots per task/checkpoint on the ten-task DMControl subset. Each root receives 20 additional independent paired repetitions. Predictors and selected actions remain fixed. Audit labels assess measurement sensitivity and are kept separate from the primary $R = 3$ measurements and calibration target.

We split the audit repetitions into $\mathcal{R}_1 = \{1, \dots, 10\}$ and $\mathcal{R}_2 = \{11, \dots, 20\}$. Each half selects a candidate, and the other half evaluates it:

$$\begin{aligned} i_a^\star(x) &\in \arg\max_{i \in \mathcal{I}} \frac{1}{10} \sum_{r \in \mathcal{R}_a} d_i^{(r)}(x), \\ \widehat{\delta}_{\mathrm{CF}}(x) &= \frac{1}{2} \sum_{a=1}^{2} \left[\frac{1}{10} \sum_{r \in \mathcal{R}_{3-a}} d_{i_a^\star(x)}^{(r)}(x) \right]. \end{aligned} \tag{26}$$

Ties favor the incumbent. We average the two directions within each root. This reference is a finite-measurement selection rule, not an oracle upper bound. Its difference from BRVL is a reference gap, not exact regret.

Selected-action diagnostics compare predicted means $\mu_{\hat{i}}$, calibrated lower scores $L_{\hat{i}}$, and independent measurements $y_{\hat{i}}$. Each panel specifies its tasks, measurement budget, and uncertainty convention.

A.10 Component Ablations

Additional ablation evaluations use Cartpole Swingup, Acrobot Swingup, Reacher Hard, Finger Turn Hard, Cup Catch, Cheetah Run, Walker Run, Hopper Hop, Quadruped Walk, and Humanoid Walk for both agent families. Shared selectors retain the full 39-task DMControl fitting scope. No replacement, BRVL, and Rel-MSE+Gate reuse main-experiment results restricted to these tasks with equal task weights.

Rel-MLP+Gate uses a capacity-matched ordinary pairwise head without enforced antisymmetry. It retains the pair features and full loss, with the incumbent score fixed at zero. It is refitted and calibrated separately. **BRVL-Mean** reuses the BRVL ensemble and maximizes μ_i over all candidates, including the zero-valued incumbent. It removes the margin but retains fallback. **Calibrated-LCB+Forced** reuses the predictor and quantile, but maximizes L_i over alternatives only. It may therefore replace the incumbent when all scores are nonpositive. Ties use a fixed candidate order. Deployment-only variants reuse saved predictions for frozen-set evaluation and have separate closed-loop runs. All variants preserve normalization, candidate generation, fitting sources, and paired evaluation conditions.

A.11 Data and Computational Efficiency

Interaction accounting. We account separately for base-agent pretraining, root acquisition, fitting and calibration branches, evaluation, and audits. Let $\mathcal{U}(x)$ contain the distinct measured interventions at root x. We define

$$C_{\text{branch}}(\mathcal{D}) = \sum_{x \in \mathcal{D}} \sum_{r=1}^{R} \sum_{u \in \mathcal{U}(x)} \ell_u^{(x,r)}, \qquad C_{\text{post}} = C_{\text{root,fit+cal}} + C_{\text{branch,fit+cal}}, \tag{27}$$

Here ℓ counts environment transitions within action repeat. It excludes physics substeps and imagined transitions. Shared incumbents count once per root and repetition. Before merging exact duplicates, each group uses 1,800 fitting, 600 calibration, and 600 frozen-evaluation branch trajectories. Actual transition counts depend on branch lengths. Reusing labels across methods or selector initializations adds compute, not interactions. Auxiliary supervision and cross-agent normalization data remain part of the accessible-data budget.

Budget and horizon sensitivity. Nested fitting-source budgets $B \in \{15, 30, 45, 60\}$ keep 20 calibration sources and evaluation sources fixed. Each budget has its own internal validation, fitted preprocessing, predictor, and calibration; fitting-plus-calibration uses $30(B + 20)$ branch trajectories before duplicate reuse. Cost axes use logged root-acquisition and branch transitions, while outcomes use native return or success units. Scoring-horizon tests use $H_{\text{plan}} \in \{1, 3, 5, 10, 15\}$ without changing candidates, native planning, execution, or real continuation. Horizon-dependent features are regenerated and predictors refitted and recalibrated; frozen-predictor transfer is a separate condition. Nested fitting budgets $B \in \{15, 30, 45, 60\}$ use the same 20 calibration sources and fixed evaluation sources. Each budget has its own internal validation, preprocessing, predictor, and calibration. Fitting and calibration require $30(B + 20)$ branch trajectories before duplicate reuse. Interaction-cost axes use logged root-acquisition and branch transitions. Outcomes use native return or success units. Scoring-horizon tests use $H_{\text{plan}} \in \{1, 3, 5, 10, 15\}$. Candidates, native planning, execution, and real continuation remain unchanged. Horizon-dependent features are regenerated, and predictors are refitted and recalibrated. Frozen-predictor transfer is treated as a separate condition.

Timing. We measure native and augmented decision paths under the same hardware, inputs, batch size, precision, warm-up, compilation, and synchronization settings. The augmented path includes incumbent generation, additional candidates, imagined features, ensemble prediction, and selection. Reused native computation is counted once. Input preparation and transfers use the same timing boundaries in both paths. Compilation and environment stepping are excluded, and compilation is reported separately. We report

$$\Delta t = \overline{T}_{\text{aug}} - \overline{T}_{\text{native}}, \qquad \text{Overhead}(\%) = 100\, \Delta t / \overline{T}_{\text{native}}. \tag{28}$$

Mean and p95 latency refer to each complete path. Component p95 values are not additive. Subtracting the two path p95 values also does not give the p95 of added latency. Stacked bars show the native mean plus the added mean, with augmented p95 shown separately. Calibration is fitted offline. Parameter counts, fitting time, calibration time, and peak memory include auxiliary modules.

A.12 Implementation Records

All tables and figures use the same unrounded records, identified by task/checkpoint, selector/member, source/split, candidate actions, repetition streams, and model/scale/calibration hashes. Records retain raw returns, success labels, effective-action mappings, and transition counts. Table 5 summarizes the selector architecture, optimization, calibration, and evaluation settings.

B Theoretical Details

All returns and value functions use the positive task scale fixed from fitting data in Eq. equation 1. The root state includes the environment state, agent or planner carry, and remaining task horizon.

Table 5: Selector, calibration, and evaluation settings. Native-agent configurations remain unchanged.

Field	Setting
Candidates and scoring	$N = 9$; $H_{\text{exec}} = 1$; one imagined rollout per candidate; $H_{\text{plan}} = 15$ for DreamerV3 and 3 for TD-MPC2.
Data split	Per task/checkpoint: 48 optimization, 12 validation, 20 calibration, and 20 frozen-evaluation roots; $R = 3$; split seed 0.
Preprocessing	Optimization-only feature standardization, with standard-deviation floor 10^{-6}; $\kappa_k = \max\{Q_{0.95} - Q_{0.05}, 1\}$ using cross-agent, equally group-weighted branch-mean returns and linearly interpolated quantiles.
Architecture	Separate task/checkpoint-specific 128-dimensional projections for z and candidate descriptors; encoder 256–256–128; pair head 256–256–1. SiLU hidden activations, linear outputs, Xavier-uniform initialization, zero biases, and no dropout.
Ensemble	$M = 5$; three complete-ensemble seeds $\{0, 1, 2\}$; shared optimization data with independent initialization and minibatch streams; no dataset bootstrapping.
Optimizer	Adam; constant learning rate 3×10^{-4}; $(\beta_1, \beta_2) = (0.9, 0.999)$; epsilon 10^{-8}; gradient-norm clipping at 10; zero optimizer weight decay.
Fitting and selection	64 complete roots per batch with equal group weights; 20,000 updates per member; validation every 500 updates. Select by validation net gain under incumbent-inclusive mean-score selection; ties favor lower harm, then earlier snapshots. One hyperparameter configuration per method.
Loss coefficients	$T = 0.1$, $\lambda_{\text{dec}} = 0.1$, $\lambda_{\text{h}} = 1$, $\epsilon_{\text{h}} = 0.05$, and explicit squared-parameter penalty $\lambda_{\text{wd}} = 10^{-5}$. Rel-MSE+Gate sets $\lambda_{\text{dec}} = 0$.
Calibration	$w_i = s_i + 10^{-3}$; $\alpha = 0.10$; 20 blocks per task/checkpoint, predictor, and ensemble seed; 19th ordered residual without interpolation. Positive-score replacement with incumbent-favoring ties.
Threshold grids	$\tau \in \{0, 0.01, \ldots, 0.20, +\infty\}$; $\lambda \in \{0\} \cup \{10^{j/4} : j = -8, \ldots, 8\} \cup \{+\infty\}$. Validation-only selection within each agent-family/benchmark pair.
Adapted references	Common ensemble size, hidden widths, optimizer, and update budget; method-specific objectives and auxiliary modules; RVL target-update rate 0.005.
Evaluation	20 paired episodes per task/checkpoint and ensemble seed; equal task weights; 10,000 paired bootstrap replicates for pointwise 95% intervals, preserving shared-selector and source dependence; bootstrap seed 0.
Measurement audit	Ten-task DMControl subset; five prespecified roots per task/checkpoint; 20 additional independent repetitions, separate from primary $R = 3$ measurements.
Timing	Common GPU; batch size 1; FP32 selector; 1,000 warm-up and 10,000 timed decisions with alternating methods and device synchronization. Report complete-path mean and p95, excluding compilation and environment stepping.

B.1 Target Semantics and Historical Context

All comparisons use the same root state, execution protocol, termination rule, and continuation policy. Under the primary protocol, $H_{\text{exec}} = 1$ and continuation follows π_b, giving

$$\Delta_i = Q^{\pi_b}(x, u_i) - Q^{\pi_b}(x, u_b).$$

Define $A^{\pi_b}(x, u) = Q^{\pi_b}(x, u) - V^{\pi_b}(x)$. For a deterministic baseline, $u_b = \pi_b(x)$ and $V^{\pi_b}(x) = Q^{\pi_b}(x, u_b)$. Hence $\Delta_i = A^{\pi_b}(x, u_i)$. For a realized action from a stochastic baseline, $\Delta_i = A^{\pi_b}(x, u_i) - A^{\pi_b}(x, u_b)$. These identities connect the target to advantage estimation (Pan et al., 2022; Pan & Schölkopf, 2026). BRVL studies the finite-candidate replacement decision, not a new definition of advantage.

Relative rollout evaluation has a classical precedent. Bertsekas (1997) studies relative Q-factor evaluation and differential training of cost-to-go differences. They aim to reduce sensitivity to simulation and approximation error. BRVL combines paired environment supervision, costs for missed improvements and harmful replacements, and simultaneous calibration of finite-rollout measurements. We test whether these components improve the gain–harm trade-off under matched information and intervention budgets.

B.2 Paired Measurement Variance

Condition on the fitting data, root, and fixed candidates. Assume the return vectors $(G_b^{(r)}, G_1^{(r)}, \dots, G_N^{(r)})$ are independent and identically distributed across repetitions. They have the specified branch marginals and finite second moments. Write $J_i = \mathbb{E}[G_i^{(r)}]$, $\Delta_i = J_i - J_b$, and $\sigma_i^2 = \mathrm{Var}(G_i^{(r)})$. Define $d_i^{(r)} = G_i^{(r)} - G_b^{(r)}$ and $y_i = R^{-1} \sum_{r=1}^{R} d_i^{(r)}$. Then $\mathbb{E}[y_i] = \Delta_i$ and

$$\mathrm{Var}(y_i - \Delta_i) = \frac{\sigma_i^2 + \sigma_b^2 - 2\,\mathrm{Cov}(G_i^{(1)}, G_b^{(1)})}{R}. \tag{29}$$

The identity follows from the variance of a difference and independence across repetitions. With fixed marginals and sample counts, positive within-pair covariance reduces variance relative to independent branches. If both marginal variances equal $\sigma^2 > 0$ and the correlation is ρ_i, paired variance is $2\sigma^2(1-\rho_i)/R$. Independent branches give $2\sigma^2/R$.

These comparisons concern measurement noise, not world-model prediction error or learned-predictor generalization. Absolute-return controls using the same paired measurements have access to the same covariance structure; pairing alone does not establish an advantage of relative regression.

For $R > 1$, the diagnostic standard error is

$$\widehat{\tau}_i = \left[\frac{\sum_{r=1}^{R} (d_i^{(r)} - y_i)^2}{R(R-1)} \right]^{1/2}.$$

Under these assumptions, $\widehat{\tau}_i^2$ is unbiased for $\mathrm{Var}(y_i)$. The standard error alone gives neither a finite-sample confidence interval nor the simultaneous measurement bounds in Eq. equation 36. Uncertainty estimates must preserve paired candidate vectors and source-block dependence because incumbent samples are shared.

B.3 Antisymmetry and Decision-Sensitive Training

Holding z and the candidate descriptors fixed, exchanging candidate and incumbent swaps the two terms in Eq. equation 3. The pair prediction therefore changes sign and vanishes on identical inputs. The ensemble mean inherits antisymmetry, while ensemble disagreement is unchanged by the exchange. Calibrated lower scores generally are not antisymmetric. Antisymmetry alone does not impose transitivity on pairwise comparisons.

For real-valued candidate scores S_i, subtracting the same score S_b preserves the maximizing set:

$$\arg\max_{i \in \mathcal{I}} S_i = \arg\max_{i \in \mathcal{I}} (S_i - S_b).$$

With the same tie rule, the selected action is therefore unchanged. Any selection benefit must arise from learning, representation, or the calibrated decision rule rather than common-score subtraction alone.

Equation equation 15 shows that each update depends on the costs and probabilities of competing candidates. This dependence remains even when the candidate's own label is fixed.

For fixed scores with a unique maximizer, the surrogate approaches that action's empirical cost as $T \to 0$. With tied maxima, it approaches the average cost of the tied actions. This limit concerns each member's unadjusted scores, not the calibrated ensemble selector. Joint regression and decision training may shift predictions away from conditional mean returns. Calibration therefore uses the final frozen predictor.

B.4 Regret from Prediction-Error Differences

Fix a root and candidate set. Define $e_i = \mu_i - \Delta_i$. Let $\bar{i} \in \arg\max_{i \in \mathcal{I}} \mu_i$ and $i^\star \in \arg\max_{i \in \mathcal{I}} \Delta_i$. Since $\mu_{\bar{i}} \geq \mu_{i^\star}$,

$$0 \leq \Delta_{i^\star} - \Delta_{\bar{i}} \leq e_{\bar{i}} - e_{i^\star} \leq \max_{i \in \mathcal{I}} e_i - \min_{i \in \mathcal{I}} e_i. \tag{30}$$

Thus $\max_{i \in \mathcal{I}} |e_i| \leq \eta$ implies regret at most 2η, without requiring correct ordering of every candidate pair. This is a deterministic bound for unadjusted mean-based selection, not a consistency

result for the training surrogate. The calibrated selector maximizes L_i rather than μ_i and is analyzed separately in Eq. equation 36.

B.5 Continuation Mismatch

Assume a constant decision-time discount $\gamma \in (0, 1]$. Fix candidate and incumbent execution prefixes of length h, with endpoint states X_h^i and X_h^b. Compare continuations π and $\tilde{\pi}$ without changing either branch's prefix law, endpoint distribution, reward convention, or task horizon. The two branches may have different endpoint distributions.

For $j \in \{i, b\}$, let J_j^π denote the expected return of branch j followed by continuation π. Cancelling the unchanged prefix rewards gives

$$J_j^\pi - J_j^{\tilde{\pi}} = \gamma^h \mathbb{E}[(V^\pi - V^{\tilde{\pi}})(X_h^j)].$$

Define $\Delta_i^\pi = J_i^\pi - J_b^\pi$. Subtracting the incumbent identity from the candidate identity yields

$$\Delta_i^\pi - \Delta_i^{\tilde{\pi}} = \gamma^h \left(\mathbb{E}[(V^\pi - V^{\tilde{\pi}})(X_h^i)] - \mathbb{E}[(V^\pi - V^{\tilde{\pi}})(X_h^b)] \right). \tag{31}$$

Consequently, if $\|V^\pi - V^{\tilde{\pi}}\|_\infty \le B$ on the relevant endpoint states, then

$$|\Delta_i^\pi - \Delta_i^{\tilde{\pi}}| \le 2\gamma^h B.$$

Termination and the finite task horizon are represented by absorbing zero-reward states. With path-dependent discounting, the corresponding branch identity uses the cumulative prefix discount inside the expectation instead of factoring out γ^h.

The primary protocol has $h = 1$, distinct from the imagined scoring horizon H_{plan}. Changing the executed prefix falls outside these identities. Repeated intervention changes continuation and visitation, so fixed-prefix bounds and finite-block measured-return guarantees alone do not certify closed-loop improvement.

C Measurement-Aware Calibration

This section distinguishes coverage of finite-rollout measurements from guarantees for expected returns. All quantities use the fitting-fixed task normalization. The label-error allowances b_i below are auxiliary conditions; they do not enter the default calibration score or deployment rule.

C.1 A Simultaneous Label-Error Construction

Condition on fitting and the block configuration fixed before label collection. Let $P \ge 1$ be the number of non-incumbent comparisons in the block. Index them by i and suppress root arguments. For each comparison, assume R_i independent repetitions with $\mathbb{E}[d_i^{(r)}] = \Delta_i$. Each difference lies in a known interval of width $W_i > 0$. The comparisons, repetition counts, and range bounds are fixed before observing the labels. For $y_i = R_i^{-1} \sum_{r=1}^{R_i} d_i^{(r)}$, define

$$b_i = W_i \sqrt{\frac{\log(2P/\beta)}{2R_i}}, \qquad 0 < \beta < 1. \tag{32}$$

Hoeffding's inequality (**?**) gives

$$\Pr(|y_i - \Delta_i| > b_i) \le 2\exp(-2R_i b_i^2 / W_i^2) = \beta/P.$$

A union bound gives simultaneous label accuracy with probability at least $1 - \beta$. Candidates and roots need not be independent. Variable block sizes are allowed if P is fixed before observing the labels; the argument then conditions on that block configuration. If $W_i = 0$, the label is exact and $b_i = 0$.

Range bounds must follow from the return definition, not sample extrema or empirical quantiles. If each normalized branch return lies in an interval of width C, the paired difference has width at most $2C$. A task scale changes the units but does not establish a range bound. Label-dependent stopping requires a sequentially valid method. These allowances can be conservative when R_i is small.

C.2 Exchangeable Blocks and Measured Coverage

Condition on fitting, validation, and model selection. Within each task/checkpoint group, the calibration blocks and a new block must be exchangeable. They use the same source sampling, root selection, candidate generation, execution, and repetition-allocation rules. The rank argument allows within-block dependence. The label-error construction separately assumes independent repetitions.

Using the score in Eq. equation 6,

$$Q_k = \max_{x \in \mathcal{B}_k} \max_{1 \le i \le N} \frac{|\mu_i(x) - y_i(x)|}{w_i(x)}, \qquad w_i(x) = s_i(x) + \varepsilon,$$

let $\ell = \lceil (n+1)(1-\alpha) \rceil$ and take $\widehat{q}$ as the ℓ-th smallest element of $\{Q_1, \dots, Q_n, +\infty\}$. The predictor, scale rule, $\varepsilon > 0$, and α are fixed before calibration. Exchangeability gives $\Pr(Q_{n+1} \le \widehat{q}) \ge 1 - \alpha$ (Angelopoulos & Bates, 2023). Ties are conservative, and $\ell = n+1$ produces an infinite quantile and incumbent fallback.

For alternatives, let $r_i = \widehat{q} w_i$ and $L_i = \mu_i - r_i$, with $\mu_b = y_b = \Delta_b = r_b = L_b = 0$. Define

$$\mathcal{E}_y = \{|\mu_i(x) - y_i(x)| \le r_i(x) \text{ for all registered } (x, i) \text{ in } \mathcal{B}_{n+1}\}.$$

Then $\Pr(\mathcal{E}_y) \ge 1 - \alpha$. This is simultaneous coverage of the measured vector, not of its unknown expectation. Calibration sample size counts source blocks rather than individual roots, candidates, or repetitions.

C.3 Marginal Expected Harm

Condition on fitting and fix a task/checkpoint group. Sample a root x from a fresh evaluation block using the stated root-selection rule. Let $\mathcal{F}$ contain the calibration data, the root, its candidates, and all information used for selection. The selected index $\hat{i}$ is $\mathcal{F}$-measurable. Assume each fresh label is integrable and satisfies

$$\mathbb{E}[y_i \mid \mathcal{F}] = \Delta_i(x), \qquad y_i \ge -B_{\mathrm{h}} \text{ almost surely}, \tag{33}$$

where $B_{\mathrm{h}} < \infty$ is a known nonnegative bound fixed conditional on fitting. Independent evaluation randomness preserves the conditional-mean requirement. The bound must follow from the return definition, not observed sample extrema.

Proposition C.1 (Marginal expected harmful loss)**.** *Under Eq. equation 8, Eq. equation 33, and incumbent-inclusive lower-score selection,*

$$\mathbb{E}[[-\Delta_{\hat{i}}]_+] \le \mathbb{E}[[-y_{\hat{i}}]_+] \le \alpha B_{\mathrm{h}}. \tag{34}$$

All expectations are conditional on fitting and marginal over calibration and a fresh evaluation root.

Proof. Since $\hat{i}$ is fixed given $\mathcal{F}$,

$$\mathbb{E}[y_{\hat{i}} \mid \mathcal{F}] = \sum_{i \in \mathcal{I}} \mathbf{1}\{\hat{i} = i\} \mathbb{E}[y_i \mid \mathcal{F}] = \Delta_{\hat{i}}(x).$$

Conditional Jensen's inequality gives

$$[-\Delta_{\hat{i}}(x)]_+ \le \mathbb{E}[[-y_{\hat{i}}]_+ \mid \mathcal{F}]\,.$$

On $\mathcal{E}_y$, coverage and incumbent fallback give $y_{\hat{i}} \ge L_{\hat{i}} \ge 0$. Otherwise, the lower bound on y_i gives $[-y_{\hat{i}}]_+ \le B_{\mathrm{h}}$. Hence

$$[-y_{\hat{i}}]_+ \le B_{\mathrm{h}} \mathbf{1}\{\mathcal{E}_y^c\}.$$

Taking expectations and using $\Pr(\mathcal{E}_y^c) \le \alpha$ proves the result. □

If all normalized branch returns lie in a common interval of width C, then $B_{\mathrm{h}} = C$ is valid. The bound controls average loss in expected return under the evaluation-root protocol. Its numerical strength depends on the known return range. It is not conditional on each realized calibration dataset and gives no pointwise or closed-loop improvement guarantee. The measurement-error allowances b_i serve a different purpose: they support simultaneous pointwise bounds on expected return differences.

C.4 Expected-Return Implications

Let $\mathcal{M}$ denote simultaneous label accuracy, $|y_i(x) - \Delta_i(x)| \le b_i(x)$, throughout the new block, with $b_b = 0$. Assume $\Pr(\mathcal{M}) \ge 1 - \beta$ and $\alpha + \beta < 1$.

Proposition C.2 (Expected-return error bound)**.** *With probability at least $1-\alpha-\beta$, every registered comparison in the new block satisfies*

$$|\mu_i(x) - \Delta_i(x)| \le r_i(x) + b_i(x). \tag{35}$$

Proof. On $\mathcal{E}_y \cap \mathcal{M}$, the triangle inequality gives $|\mu_i - \Delta_i| \le |\mu_i - y_i| + |y_i - \Delta_i| \le r_i + b_i$. A union bound gives $\Pr(\mathcal{E}_y \cap \mathcal{M}) \ge 1 - \alpha - \beta$, without requiring independence between the events. □

The default score $L_i = \mu_i - r_i$ omits the measurement allowance b_i and therefore is not, in general, a lower bound on the expected improvement Δ_i.

Proposition C.3 (Measured preservation and expected regret)**.** *Let $\hat{i} \in \arg\max_{i \in \mathcal{I}} L_i$, with incumbent-favoring ties, and let $i^\star \in \arg\max_{i \in \mathcal{I}} \Delta_i$. On $\mathcal{E}_y$, $y_{\hat{i}} \ge L_{\hat{i}} \ge 0$. On $\mathcal{E}_y \cap \mathcal{M}$, each registered root additionally satisfies*

$$\Delta_{\hat{i}} \ge L_{\hat{i}} - b_{\hat{i}} \ge -b_{\hat{i}}, \qquad 0 \le \Delta_{i^\star} - \Delta_{\hat{i}} \le 2r_{i^\star} + b_{i^\star} + b_{\hat{i}}. \tag{36}$$

Proof. The incumbent gives $L_{\hat{i}} \ge L_b = 0$. Measured coverage and label accuracy imply $\Delta_{\hat{i}} \ge y_{\hat{i}} - b_{\hat{i}} \ge L_{\hat{i}} - b_{\hat{i}}$ and $L_{i^\star} \ge \Delta_{i^\star} - b_{i^\star} - 2r_{i^\star}$. Combining these with $L_{\hat{i}} \ge L_{i^\star}$ proves the regret bound. If $\widehat{q} = +\infty$, fallback gives zero improvement, while the regret bound is vacuous when the true maximizer is an alternative. □

Nonnegative expected improvement follows on this event when $L_{\hat{i}} \ge b_{\hat{i}}$; the default positive-score gate does not enforce that condition. For a fixed additional threshold $\tau \ge 0$, let $\hat{i}_\tau$ select the largest alternative score only when it exceeds τ, otherwise retaining the incumbent. Since $L_{\hat{i}_\tau} \ge L_{i^\star} - \tau$, the corresponding regret bound is

$$\Delta_{i^\star} - \Delta_{\hat{i}_\tau} \le 2r_{i^\star} + b_{i^\star} + b_{\hat{i}_\tau} + \tau.$$

Probability scope. For a root selected from the new block under the registered sampling rule,

$$\Pr(y_{\hat{i}} < 0) \le \alpha, \qquad \Pr(\Delta_{\hat{i}} < -b_{\hat{i}}) \le \alpha + \beta.$$

The second bound controls losses larger than the measurement allowance, not all negative expected improvements. If replacement has probability $\rho > 0$, the replacement-conditional bound is at most $\min\{1, (\alpha + \beta)/\rho\}$. These are marginal bounds, not guarantees conditional on each realized calibration dataset or state.

The guarantees are marginal over calibration and a new exchangeable block. They do not certify adaptive closed-loop visitation or repeated-intervention improvement. Predictor or score selection based on calibration outcomes requires additional justification. Measurement audits and complete-episode evaluation address different aspects of reliability.

D Per-Task Learning Curves and Action-Selection Diagnostics

Base agents remain frozen. Training curves track selector optimization. Data-budget curves compare separately fitted selectors. Closed-loop tests share initial conditions but may follow different trajectories. Frozen-set diagnostics share roots, incumbents, candidates, and independent branch measurements.

Panels use a fixed task order and identify the evaluated checkpoint groups. Pointwise 95% intervals follow the paired bootstrap protocol. Resampling preserves shared-selector identities and reused evaluation sources across methods and curve points. Ensemble members are not independent evaluation units. The intervals are not simultaneous bands over a full curve. Missing evaluations are shown as gaps.

D.1 Per-Task Selector Training Curves

Checkpoint evaluation. At the full fitting budget, we save complete ensemble snapshots on an optimizer-update grid fixed before evaluation. Each point evaluates a snapshot from the same run, not a refitted selector. Each snapshot receives its own task/checkpoint quantile at $\alpha = 0.10$ using fixed calibration roots. The quantile is frozen during evaluation. Fitting sources, preprocessing, frozen candidates, and evaluation initial conditions stay unchanged. Neither calibration nor test outcomes select checkpoints. Curves mark the snapshot selected by validation.

Methods and axes. Selector-training comparisons include Abs-Reg+Gate, Rel-MSE+Gate, RVL-adapted+Gate, DAE-adapted+Gate, and BRVL. No replacement and Abs-Score appear as horizontal references because they have no selector-training trajectory. Equal update counts need not imply equal compute, especially for adapted methods. Each panel represents one task. The horizontal axis is selector optimizer updates. The vertical axis is episode return for DMControl or success rate (%) for manipulation. Curves connect evaluated checkpoints without smoothing or imposing monotonicity.

Task panels within an agent–benchmark pair evaluate the same shared selector. They do not represent separately trained task-specific selectors.

D.2 Per-Task Data-Budget Curves

BRVL, Rel-MSE+Gate, and Abs-Reg+Gate use nested budgets of $B \in \{15, 30, 45, 60\}$ fitting roots per task/checkpoint. Each budget uses a source-disjoint 80/20 optimization/validation split. Calibration and evaluation sources remain fixed. We refit preprocessing, task scales, and predictors using only the permitted fitting data, then calibrate each predictor separately. The $B = 60$ results reuse the main experiment. Tasks receive equal weight within each benchmark. We report point estimates and pointwise 95% percentile-bootstrap intervals for paired method differences using 10,000 resamples. Resampling preserves complete-selector runs across tasks and shared evaluation episodes and source blocks. Tasks, base checkpoints, and fitting and calibration data remain fixed. Across-initialization standard deviations and latency quantiles are reported separately.

D.3 Candidate-Set Examples

Figure 10 compares Abs-Score, Rel-MSE+Gate, and BRVL at two prespecified frozen roots, one for each agent family. Roots are selected without using evaluation outcomes. Each example uses a common candidate order and task-normalized scale. Using the model score S_i in Eq. equation 17, define

$$a_i(x) = S_i(x) - S_b(x), \qquad a_b(x) = 0. \tag{37}$$

TD-MPC2 candidates are scored as first actions under the common imagined continuation, not their original elite-sequence tails. Abs-Score maximizes a_i over all candidates, including the incumbent. Each learned method uses its own μ_i and $L_i = \mu_i - \widehat{q} w_i$, with $\mu_b = L_b = 0$.

The examples show these scores alongside independent measurements $y_i^{(3)}$ and mark each method's selected action. Lower scores are calibrated for the $R = 3$ measurement target, not the unknown expected return. These examples illustrate individual decisions; aggregate performance is assessed by the main experiments.

D.4 Measurement Reliability

The audit uses the prespecified ten-task DMControl subset. Five frozen roots per task/checkpoint are selected before inspecting outcomes. Each root receives 20 additional independent paired repetitions. Candidates, task scales, predictors, calibration quantiles, and selected actions remain fixed.

Audit metrics use $y_i^{(20)}$ with the same net-gain and harm definitions as the primary evaluation. The severe-harm threshold remains -0.05. Tasks receive equal weight. Any comparison across repetition budgets must use the same audit roots and selected actions. The audit replacement rate $\widehat{\rho}_{\mathrm{fr}}$ is unchanged across budgets because the selected actions are fixed.

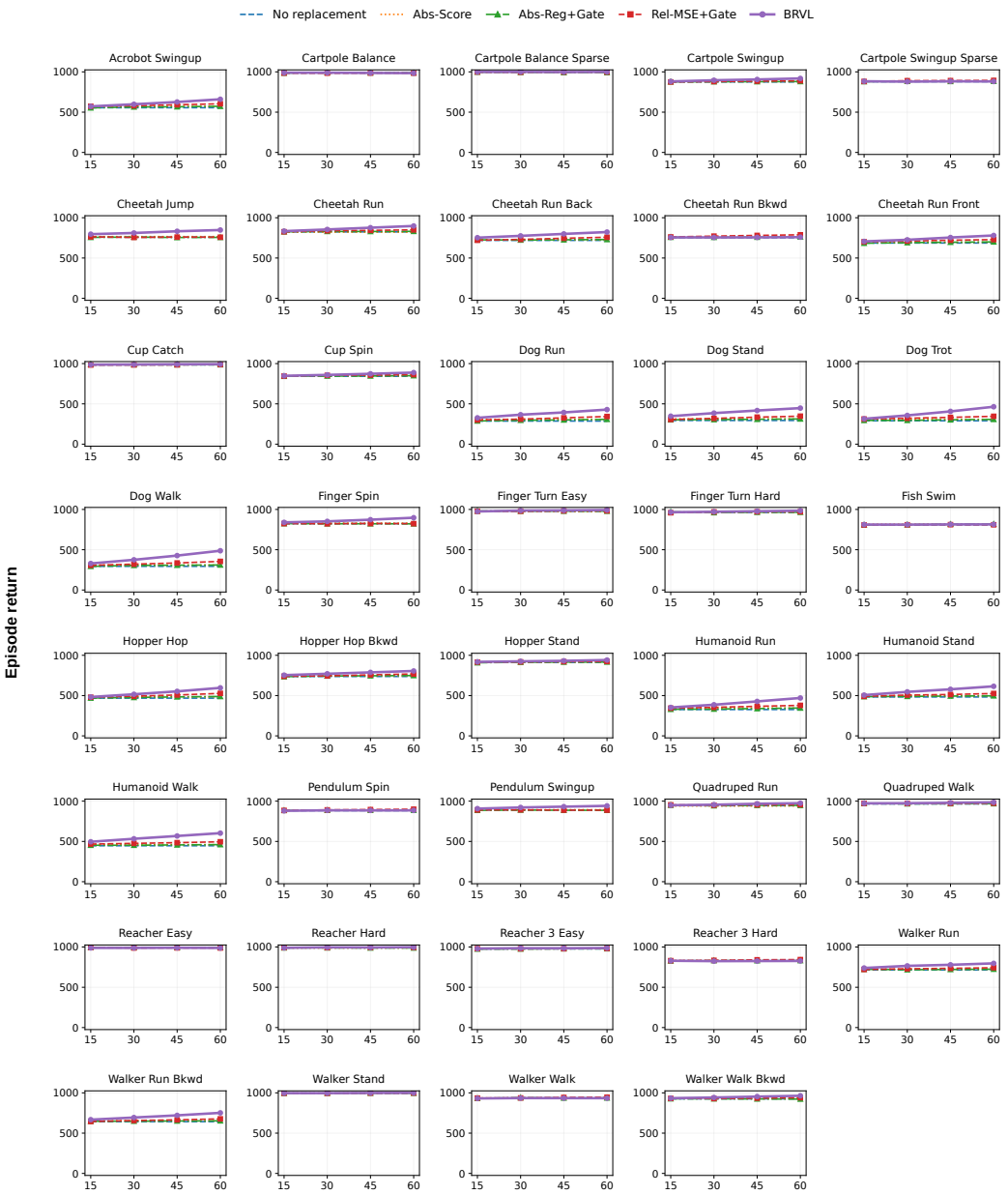


Figure 4: DreamerV3 data-budget curves on DMControl: episode return versus fitting roots per task/checkpoint.

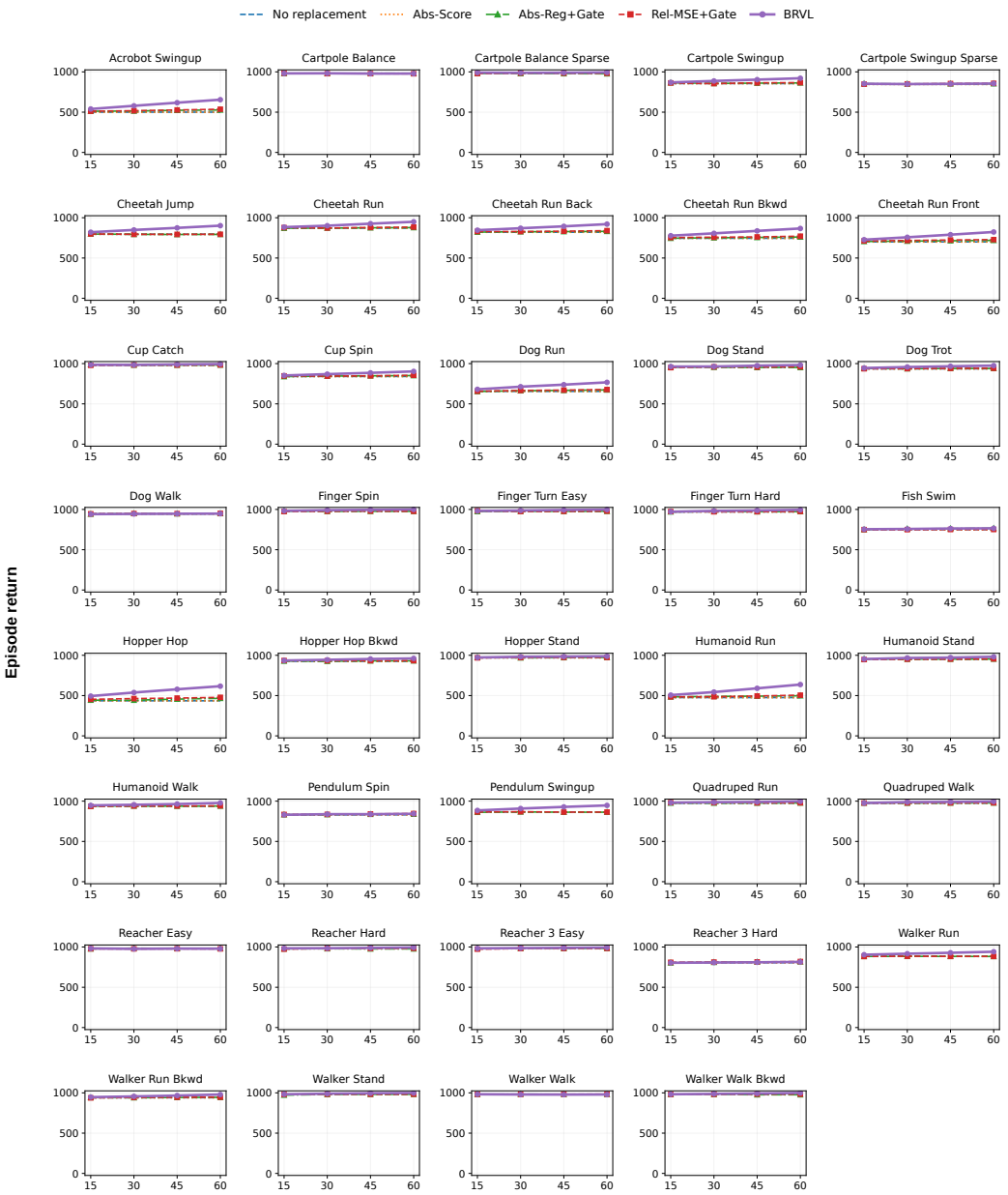


Figure 5: TD-MPC2 data-budget curves on DMControl: episode return versus fitting roots per task/checkpoint.

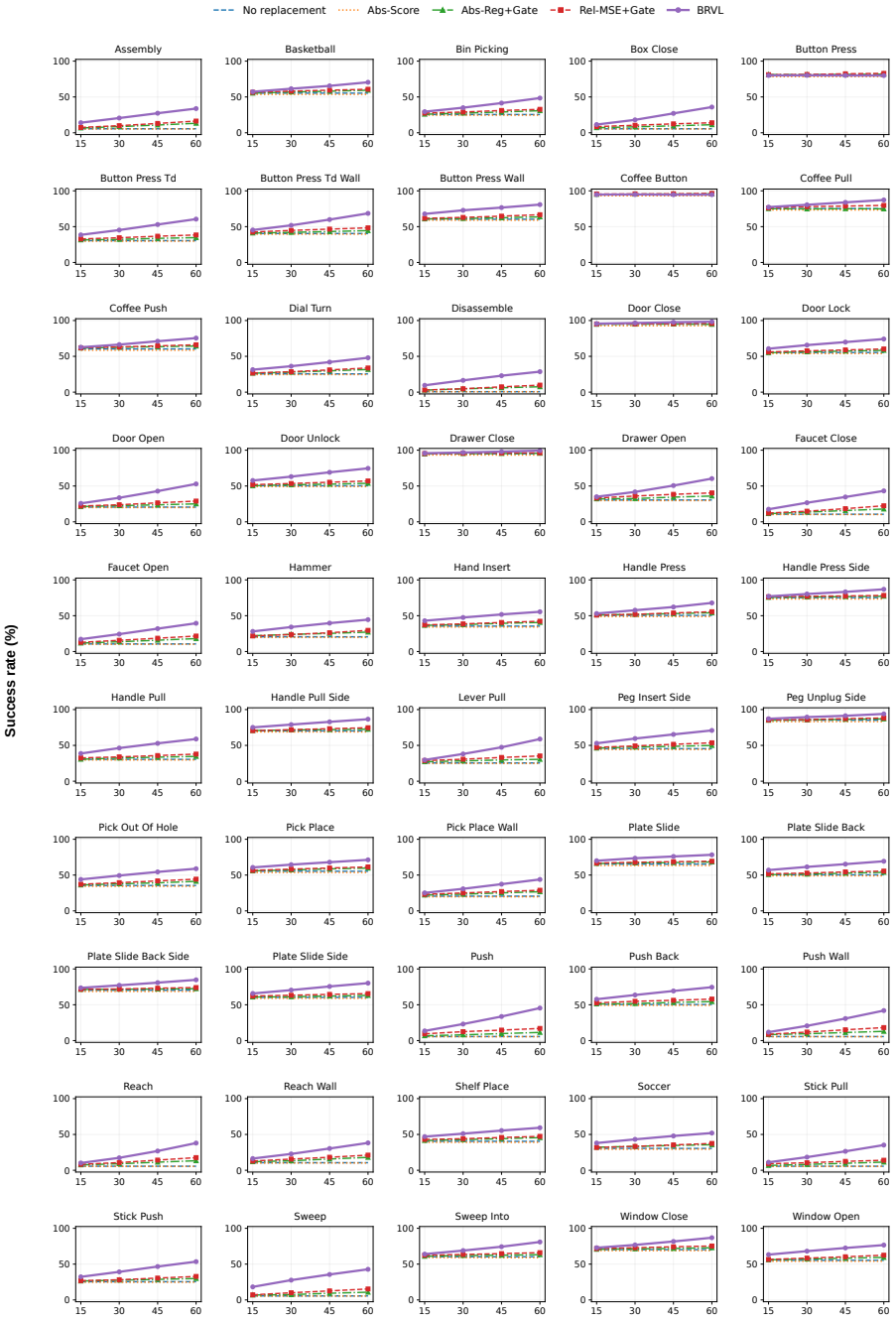


Figure 6: DreamerV3 data-budget curves on Meta-World: success rate (%) versus fitting roots per task/checkpoint.

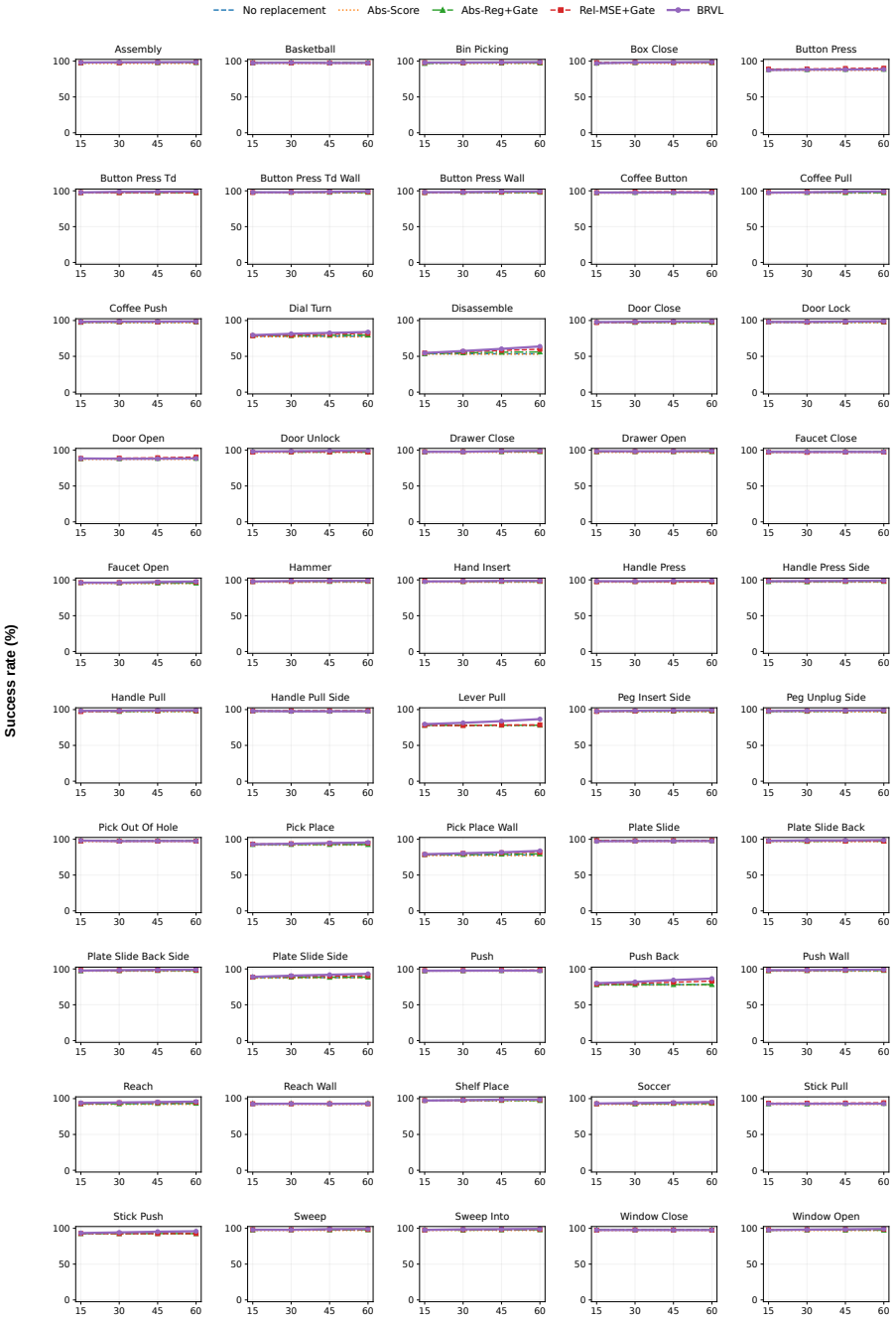


Figure 7: TD-MPC2 data-budget curves on Meta-World: success rate (%) versus fitting roots per task/checkpoint.

Audit measurements are kept separate from the primary repetitions and are not treated as exact expected returns. Primary calibration coverage concerns the $R = 3$ target. Exact deterministic replays do not add measurement information.

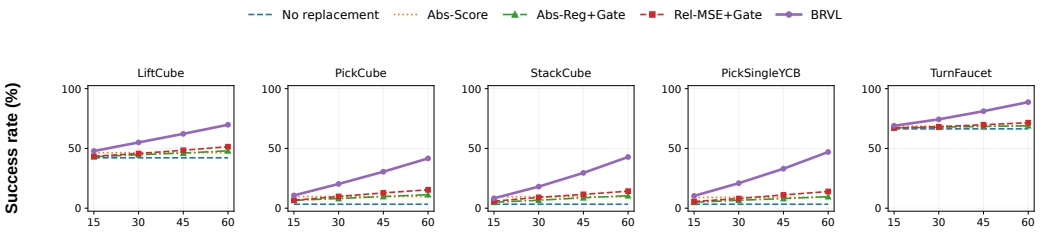


Figure 8: DreamerV3 data-budget curves on ManiSkill2: success rate (%) versus fitting roots per task/checkpoint.

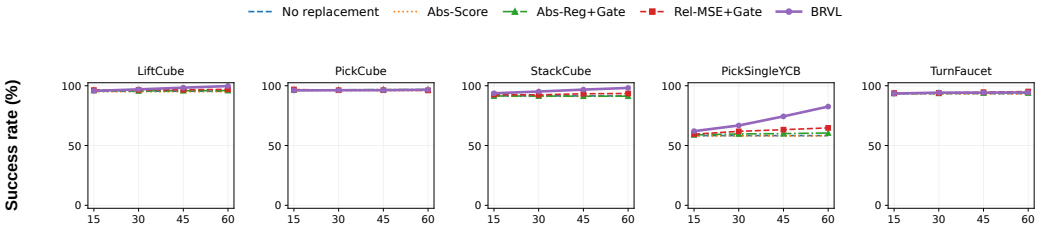


Figure 9: TD-MPC2 data-budget curves on ManiSkill2: success rate (%) versus fitting roots per task/checkpoint.

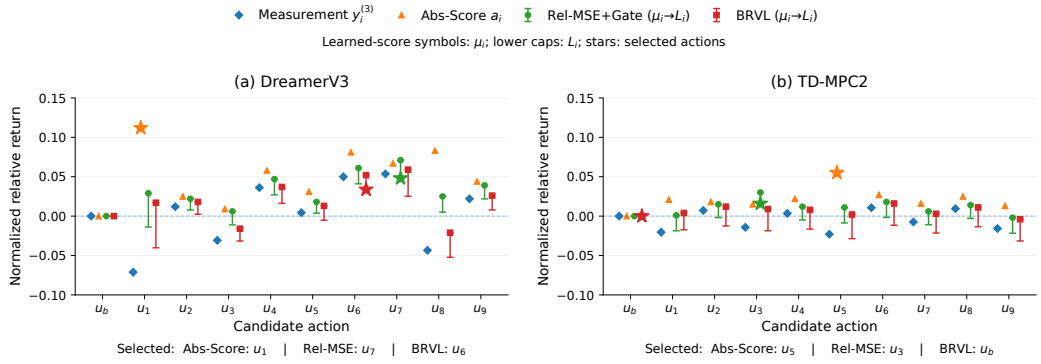


Figure 10: Candidate-set examples for DreamerV3 and TD-MPC2. Scores and independent measurements use the same candidate order and normalized units. Markers identify each method's selected action; the incumbent has score and measurement zero.