BEYOND SINGLE POINTS: CANDIDATE-SET SERVING FOR AMORTIZED MULTI-OBJECTIVE OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Amortized multi-objective optimization (MOO) accelerates test-time inference by learning a preference-conditioned mapping to solutions. Standard methods typically return a single solution per preference. However, this point-estimate approach fails when a preference admits multiple distinct high-quality solutions (leading to poor compromises due to squared-loss averaging) or when the optimal solution jumps across disconnected Pareto branches. To address this, we formalize Candidate-Set Post-training Amortized Scalarized Optimization (PASO), a precommit-then-score serving interface. Instead of a single point, a frozen model generates a sequence of candidates which are exactly scored under the queried preference at test time. Crucially, this framework allows us to theoretically and empirically decouple the effects of supervision, output cardinality, and proposal quality. Our analysis proves that while squared-loss point maps suffer from mean collapse and branch-switching obstructions, candidate sets overcome these limits—improving both the selected solution quality and the coverage of diverse alternatives. Furthermore, our theoretical bounds and experiments demonstrate that simply increasing cardinality is insufficient; candidate sets only succeed when the underlying proposal quality is high. Experiments across continuous, discontinuous, and real-world tasks confirm that candidate sets are essential for multimodal and discontinuous MOO landscapes.

1 INTRODUCTION

In many multi-objective applications, expensive training or optimization is completed before the desired trade-off is known. A user may later select a multi-task model (Navon et al., 2021), inspect an engineering design under a particular preference (Lin et al., 2022b), request a routing or knapsack solution with specified objective weights (Lin et al., 2022a), steer a fine-tuned language model toward a chosen balance of alignment objectives (Ren et al., 2025; Cheng et al., 2025), or run a control policy under a user-specified reward trade-off (Yang et al., 2025; Ambadkar et al., 2026). Preference-conditioned methods support such queries by moving most computation offline and learning to return a decision for each preference. We refer to this deployment pattern as *post-training amortized scalarized optimization* (PASO). Although evaluating a learned map at many preferences can approximate a Pareto front, its fixed-query interface usually returns only one decision.

A point interface is sufficient when one solution represents the queried scalarized problem well. It is more restrictive when the preference-to-solution correspondence is set-valued (Deb & Tiwari, 2008; Tanabe & Ishibuchi, 2020; Suzumura et al., 2026) or discontinuous (Zitzler et al., 2000). In the first case, one preference admits several separated, decision-relevant solutions with the same or nearly the same scalarized value. In the second, the preferred solution moves between disconnected branches as the preference changes, while a continuous point map must interpolate between them or remain on the wrong branch. These cases create two distinct serving goals. A system may need only the best decision it can return, in which case selected solution quality is central; alternatively, it may need to expose several high-quality choices, in which case coverage of distinct solution regions also matters.

Three mechanisms determine whether a point or a candidate set is appropriate. First, when the same preference is paired with several endpoints, squared-loss regression summarizes their conditional

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.

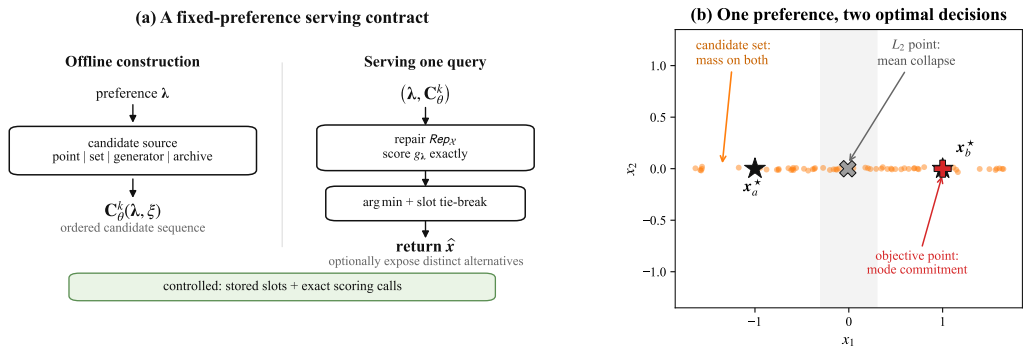


Figure 1: Candidate-set PASO at one fixed preference. A frozen source fixes an ordered sequence before common final scoring. Task repair may merge raw proposals, so native output count, stored length, and realized distinct cardinality can differ. Exact scalarized selection returns the best member; the alternative-set interface additionally exposes distinct scored candidates. Bimodal2D illustrates the separate effects of squared-loss averaging, point commitment, and output cardinality.

distribution by a feasible projection of the mean and may predict a poor compromise; direct objective training can instead commit to one good solution. Second, changing supervision does not change output cardinality: even an optimal point can represent only one separated solution region in a query. Third, increasing cardinality is useful only when the realized candidates are near-optimal and relevant to the queried preference, and when the serving rule can select among them accurately.

A small instance makes this concrete. Suppose one preference is served equally well by two separated decisions. A squared-loss regressor fitted to both returns a feasible projection of their mean, which can be far worse than either; an objective-trained map instead commits to one, optimal for the query but silent about the other; a source that emits both lets exact scoring return one and expose the other, provided both are individually near-optimal. The three services differ in supervision, in realized output count, and in what the interface can show, although each returns a decision for the same query. Figure 1(b) plots this instance and Section 4.3 states the general conditions.

We introduce *candidate-set PASO* as a *precommit-then-score serving contract* for fixed-preference deployment. Given a preference $\boldsymbol{\lambda}$, a frozen source first produces an ordered sequence of k candidates without adapting to their final query-time scores. The task repairs the candidates, evaluates their objective vectors, and scores them under the queried scalarization. The interface returns the lowest-scoring candidate and may additionally expose the distinct scored alternatives. Point maps, deterministic multi-output models, stochastic generators, and stored solution archives are interchangeable candidate sources under this contract. We study deployment settings in which scoring a modest, fixed collection is affordable but rerunning a preference-specific optimizer is not.

Many existing methods can supply candidate sources for this interface, from multi-output and generative models to preference-swept Pareto-set learners (Section 3), but they differ simultaneously in native output count, training signal, and parameterization, so comparing them as published confounds several changes at once. We therefore place them under one contract that fixes the stored budget, the repair operator, and the scoring rule, so that the source is the only free variable and each mechanism can be isolated by changing one factor at a time, and we evaluate selected quality separately from quality-gated coverage of distinct alternatives. The outcome is a regime-dependent answer rather than a ranking of methods. Our contributions are:

- We formulate candidate-set PASO as a model-agnostic, fixed-preference precommit-then-score contract that supports both single-solution serving and the exposure of distinct scored alternatives (Sections 2 and 4.1).
- We explain when a candidate set is needed. Squared-loss training on a set-valued target drives any single output toward a compromise with a quantified regret floor, one candidate can recall at most one of M separated solution regions whatever the training loss, and k candidates lower selected regret and raise coverage at a rate set by how much mass

the source places near the relevant optima; when the preferred solution switches between separated branches, every continuous point map additionally incurs a fixed regret margin on an interval of preferences (Sections 4.3 and 4.4).

- We compare point maps, deterministic sets, and stochastic generators under common stored-candidate and scoring budgets. The results show that candidate sets primarily improve high-quality alternative coverage and improve selected quality when point maps miss disconnected solution branches (Section 5).

2 Problem Setting and Serving Goals

A fixed-preference MOO query. Consider a multi-objective task in which the desired trade-off is supplied at deployment:

$$\min_{\boldsymbol{x}\in\mathcal{X}} \boldsymbol{F}(\boldsymbol{x}) = \big(f_1(\boldsymbol{x}), \ldots, f_m(\boldsymbol{x})\big), \tag{1}$$

where $\mathcal{X}$ is the feasible decision set and $\boldsymbol{F} : \mathcal{X} \to \mathbb{R}^m$ collects the m objectives. A deployment query specifies a preference $\boldsymbol{\lambda}$ in the simplex $\Delta^{m-1} = \{\boldsymbol{\lambda} \geq 0 : \mathbf{1}^\top \boldsymbol{\lambda} = 1\}$ and induces the scalarized objective $g_{\boldsymbol{\lambda}} = s_{\boldsymbol{\lambda}} \circ \boldsymbol{F}$. We consider queries for which $\mathcal{X}^\star(\boldsymbol{\lambda}) = \arg\min_{\boldsymbol{x}\in\mathcal{X}} g_{\boldsymbol{\lambda}}(\boldsymbol{x})$ is nonempty, and we denote its optimal value by $g^\star_{\boldsymbol{\lambda}}$. PASO constructs a preference-conditioned policy offline so that this query can be served without rerunning a scalarized optimizer.

Selected quality. A select-only service returns one decision $\widehat{\boldsymbol{x}}_\theta \in \mathcal{X}$ for the queried preference. We evaluate that decision by

$$r^\star_\theta = g_{\boldsymbol{\lambda}}(\widehat{\boldsymbol{x}}_\theta) - g^\star_{\boldsymbol{\lambda}} \geq 0, \quad r^{\mathrm{ref}}_\theta = g_{\boldsymbol{\lambda}}(\widehat{\boldsymbol{x}}_\theta) - g^{\mathrm{ref}}_{\boldsymbol{\lambda}}. \tag{2}$$

Here, $g^{\mathrm{ref}}_{\boldsymbol{\lambda}}$ is the best value in the policy-independent frozen set described in Appendix E.1. The frozen-reference gap may be negative because this set approximates the unknown optimum.

Alternative coverage. A select-and-expose service additionally returns a finite set $\mathcal{E}_\theta(\boldsymbol{\lambda}) \subseteq \mathcal{X}$. For each preference, we build a frozen empirical reference set from distinct high-quality solutions found by a multi-start scalarized solver: $\mathcal{M}_{\boldsymbol{\lambda}} = \{\widetilde{\boldsymbol{x}}^\star_j\}_{j=1}^{M_{\boldsymbol{\lambda}}}$. Here, $\widetilde{\boldsymbol{x}}^\star_j$ is the jth teacher-discovered alternative, $M_{\boldsymbol{\lambda}}$ is the number discovered, and $a_{\boldsymbol{\lambda}}(\boldsymbol{x})$ assigns a decision to its nearest reference alternative under the task-defined equivalence rule.

Definition 1 (Quality-gated discovered-mode recall)**.**

$$\mathrm{Rec}_{\varepsilon_{\mathrm{rec}}} = \frac{1}{M_{\boldsymbol{\lambda}}} \sum_j \mathbf{1}\big\{\exists \boldsymbol{x} \in \mathcal{E}_\theta(\boldsymbol{\lambda}) :\ a_{\boldsymbol{\lambda}}(\boldsymbol{x}) = j,\ g_{\boldsymbol{\lambda}}(\boldsymbol{x}) \leq g_{\boldsymbol{\lambda}}(\widetilde{\boldsymbol{x}}^\star_j) + \varepsilon_{\mathrm{rec}}\big\}.$$

Recall covers teacher-discovered empirical alternatives rather than an unknown complete solution set; Appendix F.1 gives the frozen construction and equivalence rules.

The two outcomes are not ordered, and neither implies the other. A service attains zero selected regret and the lowest possible recall by returning one globally optimal decision when several exist; conversely, members spread across many regions but individually loose expose a wide range of decisions while recalling none, because the quality gate discards candidates that are merely near a reference alternative in decision space. That gate prevents geometric diversity from counting as useful coverage, which is why we report the two quantities side by side rather than combining them. These metrics specify the two fixed-preference serving goals. Section 4.1 defines how candidate-set PASO constructs and scores the returned outputs.

3 Related Work

Learned preference maps. Navon et al. (2021) train a hypernetwork to map each preference to a solution, and Lin et al. (2022b) learn the corresponding preference-to-solution map for expensive multi-objective problems. These methods approximate a Pareto front by evaluating the learned map at many preferences, while typically returning one solution for each fixed $\boldsymbol{\lambda}$. Recent work extends amortization across tasks or problem instances (Wei et al., 2026; Cheng et al., 2026), whereas Bayesian and in-context optimization propose new evaluations sequentially (Paria et al., 2020; Zhang et al., 2026). Our setting treats each deployment preference as a fixed query and asks what the serving system should return for that query.

Populations and multimodal solutions. Decomposition, particle, hypervolume, and evolutionary methods construct populations across objectives or subproblems (Zhang & Li, 2007; Liu et al., 2021; Deist et al., 2021; Deb et al., 2002). Multimodal MOO additionally seeks decision-space diversity among solutions with similar objective values (Deb & Tiwari, 2008; Tanabe & Ishibuchi, 2018; 2020; Suzumura et al., 2026). These approaches target global-front search, whereas candidate-set PASO evaluates several decisions for one frozen scalarization.

Multiple outputs and best-of-k selection. Mixture-density, conditional generative, and multiple-choice models return several outputs per input (Bishop, 1994; Sohn et al., 2015; Guzman-Rivera et al., 2012; Lee et al., 2016). Learned combinatorial and generative optimizers similarly select among several rollouts or designs (Kwon et al., 2020; Lin et al., 2022a; Jain et al., 2023; Bahlous-Boldi et al., 2026). Candidate-set PASO evaluates these sources under one fixed-query rule, separating selected quality from coverage of distinct high-quality alternatives.

What remains open. Each of these lines supplies a candidate source, and several already report best-of-k or population-level results. What is not settled for fixed-preference serving is the interface question: when the source must commit its candidates before the query is scored, how many candidates it should return, what the returned object should be, and how the resulting service should be measured. We address that question directly and compare sources drawn from all three lines under one stored-candidate budget, one repair operator, and one scoring rule.

4 Candidate-Set PASO: Contract, Sources, and Mechanisms

4.1 The Precommit-Then-Score Serving Contract

Candidate-set PASO is a model-agnostic serving contract that accepts different candidate generators. Offline, a preference-conditioned source is trained or assembled and then frozen; for a deployment preference $\boldsymbol{\lambda}$ it fixes a complete ordered sequence before final scoring, and the task then repairs and evaluates every member, returns the lowest-scoring repaired decision, and may additionally expose the distinct scored alternatives.

Let $[k] = \{1, \dots, k\}$. A frozen source, together with a fixed expansion or truncation rule, produces

$$\mathbf{Z}_\theta^k(\boldsymbol{\lambda}, \xi) = (\boldsymbol{z}_\theta^{(1)}, \dots, \boldsymbol{z}_\theta^{(k)}),$$

where ξ denotes optional source randomness. Precommitment means that this complete sequence is fixed before the evaluator scores any newly proposed decision. The source may condition on $\boldsymbol{\lambda}$, sample from a frozen model, or rank archive entries by cached objective vectors; a score obtained for a new proposal cannot be used to construct later slots.

A task-defined single-valued repair produces the stored sequence

$$\boldsymbol{x}_\theta^{(j)} = \mathsf{Rep}_{\mathcal{X}}(\boldsymbol{z}_\theta^{(j)}), \qquad \mathbf{C}_\theta^k = (\boldsymbol{x}_\theta^{(1)}, \dots, \boldsymbol{x}_\theta^{(k)}) \in \mathcal{X}^k.$$

Here, $\mathcal{X}^k$ is the k-fold Cartesian product: order and repeated entries are retained. The task-owned evaluator then computes $v_j = s_{\boldsymbol{\lambda}}(\boldsymbol{F}(\boldsymbol{x}_\theta^{(j)}))$. The number of distinct repaired decisions is $q_\theta = |\{\boldsymbol{x}_\theta^{(j)} : j \in [k]\}| \le k$. The select-only decision is

$$\widehat{j}_\theta = \min \arg\min_{j \in [k]} v_j, \qquad \widehat{\boldsymbol{x}}_\theta = \boldsymbol{x}_\theta^{(\widehat{j}_\theta)}, \tag{3}$$

The outer minimum breaks ties by taking the earliest index in the stored sequence, which occurs when repair produces duplicates, so the returned decision is deterministic and single-valued. For select-and-expose serving, exact duplicates are removed in first-occurrence order to form $\mathcal{E}_\theta(\boldsymbol{\lambda})$; the task equivalence used by recall may additionally identify representation-equivalent decisions.

Figure 1(a) shows the resulting flow, and Algorithm 1 in Appendix C writes it out as pseudocode.

4.2 Candidate Sources, Supervision, and Resources

Before any repetition or truncation, n_{nat} is the number of candidates that the frozen source returns for one query: 1 for a point map, J for a J-member ensemble or J-head model, k for k generator

Table 1: Four candidate sources written directly in the common ordered-sequence notation, with n_{nat} the native output count. Every sequence is fixed before repair and final scoring.

Source	Precommitted raw sequence $\mathbf{Z}_\theta^k$
Point map	$\mathbf{Z}_\theta^k = \mathsf{T}_k((\boldsymbol{z}_\theta(\boldsymbol{\lambda}))), \quad n_{\mathrm{nat}} = 1$
Deterministic set	$\mathbf{Z}_\theta^k = \mathsf{T}_k\big((\boldsymbol{z}_{\theta,r}(\boldsymbol{\lambda}))_{r=1}^J\big), \quad n_{\mathrm{nat}} = J$
Conditional generator	$\mathbf{Z}_\theta^k(\boldsymbol{\lambda}, \xi_{1:k}) = (\mathsf{Gen}_\theta(\boldsymbol{\lambda}, \xi_j))_{j=1}^k, \quad n_{\mathrm{nat}} = k$
Archive retrieval	$\mathbf{Z}_\theta^k = \mathsf{T}_k\big(\mathrm{sort}_{g_{\boldsymbol{\lambda}}}^{\uparrow}((\boldsymbol{a}_r)_{r=1}^N)\big), \quad n_{\mathrm{nat}} = \min(k, N)$

samples, and $\min(k, N)$ for retrieval from an archive of N entries. A fixed, score-independent rule T_k converts this native output to k stored slots by truncating or repeating entries in order. Repair then leaves $q_\theta \le k$ distinct decisions.

The four source classes in Table 1 differ in how they obtain candidates. Direct objective training optimizes a point or multi-output map through $g_{\boldsymbol{\lambda}}$, whereas endpoint supervision runs a scalarized solver offline from several starting decisions and retains the endpoints whose values are close to the best one found for that preference. A learner can fit all retained endpoints, fit representatives obtained by grouping nearby endpoints, or store those representatives in an archive; this supplies multiple plausible targets without treating the teacher's discovered set as the complete optimum set. For an archive $\mathcal{A} = \{(\boldsymbol{a}_r, \boldsymbol{F}(\boldsymbol{a}_r))\}_{r=1}^N$, $\mathrm{sort}_{g_{\boldsymbol{\lambda}}}^{\uparrow}$ denotes stable ascending ordering by scalarized values computed from the cached objective vectors.

The common task-owned scorer serves both interfaces. For select-only serving it converts within-sequence variation into a returned decision; for select-and-expose serving it supplies the quality gate that prevents geometric diversity from being counted as useful coverage. Appendix F.6 studies what changes when the final scores are imperfect.

4.3 Fixed-Preference Mechanisms

We now state the three mechanisms formally. Throughout this subsection, fix $\boldsymbol{\lambda}_0$, write $g(\boldsymbol{x}) = s_{\boldsymbol{\lambda}_0}(\boldsymbol{F}(\boldsymbol{x}))$, and let $g^\star = \min_{\boldsymbol{x}\in\mathcal{X}} g(\boldsymbol{x})$ for continuous g on nonempty compact $\mathcal{X}$.

Supervision: squared-loss mean collapse. Suppose the same preference is paired with a distribution of valid teacher solutions.

Lemma 1 (Constrained squared-loss Bayes actions)**.** *Let $\boldsymbol{Y} \sim \mu$ have finite second moment and mean $\boldsymbol{m}_\mu$. The complete set of constrained squared-loss Bayes actions is $\mathcal{A}_\mu = \arg\min_{\boldsymbol{a}\in\mathcal{X}} \mathbb{E}\|\boldsymbol{a} - \boldsymbol{Y}\|_2^2 = \mathrm{Proj}_{\mathcal{X}}(\boldsymbol{m}_\mu)$. If $\Delta_\mu := \inf_{\boldsymbol{a}\in\mathcal{A}_\mu}[g(\boldsymbol{a}) - g^\star] > 0$, every Bayes action incurs regret at least Δ_μ. For $\mu = \frac{1}{2}(\delta_{\boldsymbol{x}_a} + \delta_{\boldsymbol{x}_b})$, if the midpoint $\boldsymbol{m}_\mu = (\boldsymbol{x}_a + \boldsymbol{x}_b)/2$ is feasible, then $\mathcal{A}_\mu = \{\boldsymbol{m}_\mu\}$ and its regret is $g(\boldsymbol{m}_\mu) - g^\star$.*

This is a target-and-loss effect, not a limitation of every single-output policy. An objective-trained point map can instead commit to one minimizer and attain low selected regret. For Bimodal2D, the midpoint between the two symmetric optima has exact scalarized regret $1/8$, providing a concrete instance of the lemma (Appendix D).

Realized cardinality: the one-candidate ceiling. Objective training can remove averaging without increasing the number of alternatives returned.

Lemma 2 (Single-candidate recall cap)**.** *Fix M distinct decision-relevant global minimizers $\{\boldsymbol{x}_j^\star\}_{j=1}^M$ and pairwise-disjoint membership cells $\{D_j\}_{j=1}^M$. For one realized candidate $\boldsymbol{x}$, let $I_j^{\varepsilon_{\mathrm{th}}}(\boldsymbol{x}) = \mathbf{1}\{\boldsymbol{x} \in D_j,\ g(\boldsymbol{x}) \le g(\boldsymbol{x}_j^\star) + \varepsilon_{\mathrm{th}}\}$. Then $\sum_j I_j^{\varepsilon_{\mathrm{th}}}(\boldsymbol{x}) \le 1$, so every single-candidate policy has $\mathrm{Rec}_{\varepsilon_{\mathrm{th}}} \le 1/M$ per realization and in expectation, with zero recall when the quality gate fails.*

The ceiling is independent of the training loss and applies even when the returned point is globally optimal.

Proposal quality: what k candidates can provide. Increasing cardinality removes the mechanical ceiling only when the source reaches the relevant regions.

Theorem 1 (Candidate-set quality and coverage)**.** *Let $B_j = B_\rho(\boldsymbol{x}_j^\star) \cap \mathcal{X} \subseteq D_j$ be pairwise disjoint, and suppose every $\boldsymbol{x} \in \cup_j B_j$ satisfies $g(\boldsymbol{x}) - g^\star \le b_\rho$. Let the final post-repair proposal law Q obey $q_j := Q(B_j) \ge p_j > 0$ and $q_\cup := Q(\cup_j B_j) = \sum_j q_j$. Draw k candidates conditionally i.i.d. from Q and select by Equation* (3)*; let $R_{\max} = \sup_{\mathcal{X}}[g - g^\star] < \infty$. Then $\mathbb{E}[g(\widehat{\boldsymbol{X}}_k) - g^\star] \le b_\rho + (1 - q_\cup)^k R_{\max} \le b_\rho + (1 - \sum_j p_j)^k R_{\max}$. Whenever $b_\rho \le \varepsilon_{\mathrm{th}}$, $\Pr(\mathrm{Rec}_{\varepsilon_{\mathrm{th}}} = 1) \ge [1 - \sum_j (1 - p_j)^k]_+$ and $\mathbb{E}[\mathrm{Rec}_{\varepsilon_{\mathrm{th}}}] \ge M^{-1} \sum_j [1 - (1 - p_j)^k]$. If the first upper bound is below Δ_μ, it certifies lower expected regret than every squared-loss Bayes action in Lemma 1.*

Selected quality depends on hitting any near-optimal region, whereas full recall depends on hitting every desired region. Inverting the two bounds gives sufficient budgets $k \ge \log(R_{\max}/(\epsilon - b_\rho))/[-\log(1 - q_\cup)]$ for a selected-regret target ϵ, and $k \ge \log(M/\delta)/[-\log(1 - p_{\min})]$ for full recall with probability $1 - \delta$ when $p_j \ge p_{\min}$. Both grow only logarithmically in the target but blow up as the hit probabilities vanish, so a budget is meaningful only together with proposal quality. Conditional independence is needed for these probability statements, not for the realized-hit implication: any repaired sequence containing a member of $\cup_j B_j$ has selected regret at most b_ρ, and one hitting every B_j has full recall when $b_\rho \le \varepsilon_{\mathrm{th}}$. Proofs and sufficient regularity conditions for b_ρ are in Appendix D.

4.4 Branch Switching and Point-versus-Set Guidance

The preceding results hold at one preference. A second obstruction arises across preferences when the preferred solution changes between separated branches.

Theorem 2 (Endpoint and separator obstruction for continuous point maps)**.** *Let $\boldsymbol{\lambda} : [a, b] \to \Delta^{m-1}$ be continuous, let $\mathcal{X}$ be compact, let $g_{\boldsymbol{\lambda}(s)}(\boldsymbol{x})$ be jointly continuous on $[a, b] \times \mathcal{X}$, and write $r(s, \boldsymbol{x}) = g_{\boldsymbol{\lambda}(s)}(\boldsymbol{x}) - g^\star_{\boldsymbol{\lambda}(s)}$. Let $N_A, N_B \subset \mathcal{X}$ be disjoint and closed, and assume for some $\gamma > 0$ that $r(a, \boldsymbol{x}) \ge \gamma$ outside N_A, $r(b, \boldsymbol{x}) \ge \gamma$ outside N_B, and $r(s, \boldsymbol{x}) \ge \gamma$ outside $N_A \cup N_B$ for every $s \in [a, b]$. Then every continuous point map $h : [a, b] \to \mathcal{X}$ satisfies $\max_s r(s, h(s)) \ge \gamma$ and incurs regret at least $\gamma/2$ on a nonempty open subinterval of (a, b).*

The hypotheses constrain only the endpoint preferences and the separator between the branches, so competing branch values may cross continuously; the obstruction is continuity of the map, not the single-output interface. A candidate set avoids it when its realized members cover the competing branches and exact scoring selects the member appropriate for the query. Appendix D gives the proof, a closed-form instance, and a numerical check that ZDT3 satisfies the hypotheses with $\gamma = 10^{-4}$ at each of its four branch switches under the experimental scalarizer.

Together the four results give a practical rule: a point policy suffices when only one decision is required and an objective-trained map attains the needed quality; a candidate set is called for when several task-distinct alternatives are valued at one preference or when a continuous point map misses disconnected branches, and it helps only if its source produces near-optimal, preference-relevant candidates that the selector can tell apart.

5 Experiments

The experiments ask when each mechanism of Section 4 actually governs a serving outcome. Each subsection changes one factor at a time under the common ledger: supervision at fixed cardinality, cardinality at fixed supervision, and candidate placement at fixed cardinality. We then ask when a single point is already enough and what a set costs.

5.1 Experimental Setting

Tasks. Four headline tasks are chosen so that each mechanism is identifiable rather than confounded. Bimodal2D places two separated optima at the symmetric preference, so squared-loss averaging is exactly measurable: their midpoint has a known scalarized regret. OmniTest keeps that fixed-preference multiplicity but supplies many decision-space modes (Deb & Tiwari, 2008),

Table 2: Selected frozen-reference gap and quality-gated discovered-mode recall (means over ten seeds) under 64 stored slots and 64 exact scoring calls; digit-pair uses signed-permutation quotient recall. Native output counts and source-specific costs differ. Appendix E reports uncertainty, complete configurations, and additional tasks.

		Bimodal2D		OmniTest		ZDT3		Digit-pair	
Family	Source	Ref. gap ↓	Recall ↑	Ref. gap ↓	Recall ↑	Ref. gap ↓	Recall ↑	Ref. gap ↓	Q-recall ↑
Single candidate	Endpoint (L_2)	0.1145	0.000	0.0715	0.006	0.0495	0.136	0.5705	0.000
	PHN-obj	**-4.5e-10**	0.500	3.8e-4	0.105	0.0369	0.223	-0.0447	0.756
Deterministic set	PHN-obj ens.-8	**-4.6e-10**	**1.000**	**1.9e-4**	0.121	0.0339	0.370	**-0.0453**	**0.964**
	MCL-8	**1.6e-7**	**1.000**	2.9e-4	**0.441**	0.0476	0.131	0.0970	0.043
Conditional generator	MDN	1.2e-4	0.922	9.1e-4	0.148	0.0073	0.186	0.1224	0.071
	Endpoint+noise (σ=1)	6.5e-5	0.949	0.0107	0.015	**0.0069**	**0.604**	0.5047	0.000

making coverage rather than averaging the binding constraint. ZDT3 instead moves its preferred solution between disconnected Pareto-front branches as the preference changes (Zitzler et al., 2000), the setting of Theorem 2. Digit-pair multi-task learning repeats the multiplicity question when a decision is the full weight vector of a shared-trunk classifier trained on real images, where alternatives are meaningful only modulo the network's own symmetries. Appendix E reports the Adult fairness–accuracy pair and the complete result matrix.

Candidate sources. Two single-candidate policies differ only in supervision: Endpoint regresses the preference onto teacher endpoints under squared loss, whereas PHN-obj is a preference hypernetwork trained directly on $g_{\boldsymbol{\lambda}}$ with no teacher targets. Three candidate-set sources then vary cardinality and placement: PHN-obj ens.-8 keeps objective training and changes only the realized output count through eight independently seeded maps; MCL-8 is one network with eight heads trained by winner-takes-all imitation of the teacher endpoints; and MDN fits a mixture density to the near-endpoint law and draws from it. We deliberately add Endpoint+noise, the crudest possible set—the Endpoint map plus Gaussian draws at a teacher-derived per-dimension scale, fixed at $\sigma = 1$—because it varies placement while adding no modelling capacity. Appendix B gives every target, loss, and native output count, and Appendix F.2 evaluates non-parametric retrieval from the offline teacher archive under the same ledger.

Common ledger and metrics. The comparison evaluates 64 frozen test preferences and gives each query 64 stored slots and 64 final objective evaluations, so no source can buy quality with extra scoring calls. Only these two interface resources are controlled: native output count, model evaluations, latency, parameter storage, and offline computation remain source-dependent and are reported in Appendices F.2 and F.5, so this is a common-interface comparison rather than an end-to-end cost comparison. All runs use normalized smooth Tchebycheff scalarization with $\tau = 0.05$, with configurations selected on validation preferences and then frozen. The frozen-reference gap r_θ^{ref} measures the returned decision and quality-gated recall with $\varepsilon_{\mathrm{rec}} = 10^{-3}$ measures coverage; for neural-network decisions recall is computed in the signed-permutation quotient of Appendix F.3, so two weight vectors representing the same function are not counted as two alternatives. Each headline task uses seeds 0–9 with the seed as the statistical unit, and Table 2 reports seed means; standard deviations, seed-paired intervals, and the full matrix are in Appendix E.

5.2 Fixed-Preference Alternatives: Candidate Sets Improve Coverage

Supervision decides whether a point averages or commits. Bimodal2D isolates the first mechanism, and the two single-candidate rows of Table 2 behave as Lemma 1 predicts: the squared-loss regressor settles between the two separated optima and recalls neither, with a selected gap close to the analytic midpoint regret of $1/8$. Changing only the training signal removes the failure entirely, and the objective-trained map reaches numerically zero gap without any change in output cardinality.

Cardinality decides how many alternatives can be exposed. That same map recalls exactly one of the two regions, the ceiling of Lemma 2 at $M = 2$, and no further training can raise it because the bound does not depend on the loss. Holding supervision fixed and giving the source eight independently seeded outputs lifts recall to 1.000 while leaving the selected gap unchanged, and

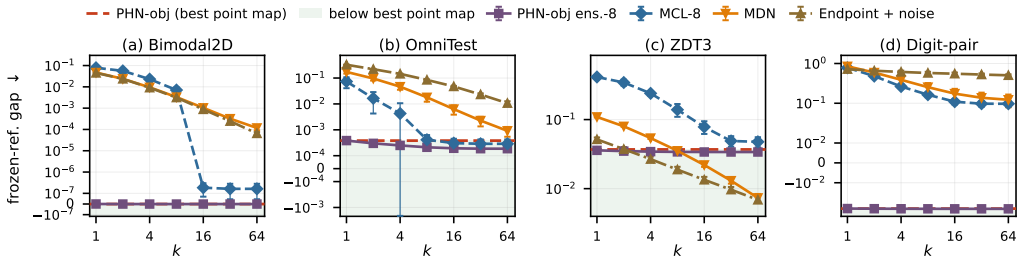


Figure 2: Selected gap on random subsets of the stored sequences for the four headline tasks: for each stored 64-slot sequence and each $k < 64$, the gap is recomputed over 20 uniform size-k subsets without replacement, so $k = 64$ reproduces Table 2. Markers are seed means with approximate 95% normal intervals. The dashed line is the strongest learned point map and the shaded band the region below it; axes are symmetric-log because that baseline is slightly negative on (a) and (d). Native lower-k training and latency are unchanged.

digit-pair reproduces this on real data in the symmetry quotient: the ensemble moves the gap by less than 10^{-3} but raises recall from 0.756 to 0.964. The two outcomes are therefore controlled by different design choices.

Cardinality alone is not sufficient. OmniTest, which has many modes rather than two, separates the candidate-set sources from each other. Every set source there fills the same budget, yet their recalls span more than an order of magnitude and the ordering does not follow model capacity: the multi-head set, whose winner-takes-all training makes its heads specialize on different endpoints, recalls the largest share, whereas the ensemble that was best on the two-mode task barely improves on its own point baseline. Since their selected gaps all stay on the 10^{-4} scale, the difference lies entirely in where the candidates land—the premise $q_j > 0$ of Theorem 1 rather than the budget k. No learned source we tested recalls a majority of OmniTest's alternatives—only retrieval from the stored teacher archive does, at 0.960 (Appendix F.2)—so a suitable learned set raises coverage substantially without resolving it.

5.3 Discontinuous Branches: Candidate Sets Improve Selected Quality

ZDT3 changes the question from what the interface can expose to what it can return. Its preferred decision switches between disconnected Pareto-front branches as the preference varies, which is the obstruction of Theorem 2 for continuous point maps, and it is the only headline task on which a candidate set improves the selected decision. Two sources of very different sophistication achieve that, both by roughly a factor of five over the strongest learned point map: a learned mixture density, and a fixed-scale Gaussian perturbation of the very squared-loss map that collapses elsewhere. That the crude perturbation matches the learned generator is our clearest evidence that the operative requirement is candidate placement, not modelling capacity. The task also supplies its own negative control: MCL-8 stores the same number of candidates and its heads do specialize, yet it stays worse than the single point map because they spread within a branch rather than across branches, and the objective-trained ensemble improves only marginally because it inherits the point map's supervision. Candidate count is therefore not the operative guarantee, and here the benefit appears in the returned decision rather than in recall. What is obstructed is continuity rather than the single-output interface itself: nearest-preference archive lookup returns one decision, is discontinuous in the preference, and reaches 2.4×10^{-4} under the same ledger (Appendix F.2).

5.4 Point Sufficiency, Candidate Budget, and Cost

When one point is enough. A candidate set offers little selected-quality benefit when an objective-trained point policy already follows a well-behaved preference–solution correspondence: on the Adult linear task every source attains a gap near zero (Appendix E). Digit-pair gives the complementary case, where one point is sufficient for the returned decision but not for exposing the

reference alternatives. The small negative gaps of the objective-trained maps in Table 2 are train-split optimism rather than evidence of beating the optimum: re-scoring the same stored decisions on held-out data turns them positive on both weight-space tasks while preserving every ordering between supervision types and between cardinalities (Appendix F.4).

What the budget buys. Figure 2 recomputes the selected gap on random size-k subsets of the trained 64-slot sequences for every headline task. Only on ZDT3 does any source cross materially below the best point map, and there two of them do so at small budgets; on Bimodal2D and OmniTest the point map already sits at the achievable floor, and on digit-pair no source approaches it at any budget. Every curve improves monotonically with k, as exact selection over a longer stored sequence must, but monotone improvement is not crossing: MCL-8 gains nearly an order of magnitude on ZDT3 and still ends worse than one well-trained point map.

What the set costs. Under the controlled $k = 64$ rescoring ledger the interface change is inexpensive on these tasks: moving from a point map to the default MDN changes median end-to-end time from 0.17 to 0.23 ms on ZDT3 and from 20.5 to 22.2 ms on Digit-pair. The relative overhead is largest where the objective is cheap and smallest where evaluating $\boldsymbol{F}$ dominates. These timings include the common 64-call scorer; Appendix F.5 separates native generation, storage, ranking, and latency, and puts the break-even against per-query solving in the low hundreds of queries.

Which source to use. The results order the questions to ask rather than rank sources: if one decision suffices and an objective-trained point map is available, use it. If alternatives matter, the source must place candidates in distinct near-optimal regions—an independently seeded ensemble is reliable once the point map is good, whereas head specialization covers more regions when many exist—and if the preferred decision jumps between branches, dispersion helps and perturbing an existing map is the cheapest option. Where feasibility is a hard requirement rather than a repaired coordinate, cardinality decides whether the service can answer at all: under a feasibility-first selector on the constrained RE24 design task, the MDN set returns a satisfying design for 84% of preferences, whereas the two point maps do so for about 2% at best (Appendix F.7).

6 Limitations and Broader Impact

Candidate-set serving needs a meaningful decision-space equivalence and query-time scoring accurate enough to rank a finite sequence, so noisy or expensive objectives can shrink the usable budget and change the selected member: with one percent selector noise the sets keep their margin over the collapsed squared-loss map on every task, whereas ten percent erases their advantage on ZDT3, where the gap is smallest (Appendix F.6); Appendices F.3 and F.7 examine symmetry and feasibility-first selection. Coverage is measured against teacher-discovered alternatives, the resource ledger matches stored slots and scoring calls rather than end-to-end cost, and the formal results hold under the conditions of Appendix A. Exposing several scored alternatives makes trade-offs visible but does not determine which should be chosen, and it inherits the objectives, data, and equivalence rules behind the candidates.

7 Conclusion

This paper treats amortized multi-objective optimization as a serving-interface choice: candidate-set PASO asks a frozen source to precommit a finite sequence, exactly scores it for the queried preference, and returns the best member while optionally exposing the distinct alternatives. Placing heterogeneous sources under that one contract separates three effects that point-versus-set comparisons otherwise conflate: squared-loss supervision can average incompatible endpoints, one realized output cannot cover several disjoint solution regions, and extra candidates help only when the source supplies preference-relevant near-optimal decisions. The answer is regime-dependent: where one preference admits several good decisions a suitable set mainly buys alternative coverage, where the preferred decision jumps between disconnected branches it also improves the returned decision, and where the correspondence is well behaved one objective-trained point is enough. What a set cannot do is repair a source whose candidates land in the wrong place, which is why proposal quality, not cardinality, is the design variable this contract exposes.

Ethics Statement

This work introduces no new human-subject data collection.

Reproducibility Statement

All experiments use fixed random seeds, frozen test preferences, and configurations selected on validation preferences before test evaluation. Appendices E and E.1 provide the task specifications, training settings, seed counts, reference construction, complete result matrices, hardware, and timing protocol.

The Use of Large Language Models

Large language models assisted with copy-editing, routine code refactoring, and experiment planning; the authors verified all scientific claims and reported artifacts, and language models are neither components of the method nor part of any experiment.

References

Tanmay Ambadkar, Sourav Panda, Shreyash Kale, Jonathan Dodge, and Abhinav Verma. Preference conditioned multi-objective reinforcement learning: Decomposed, diversity-driven policy optimization. *arXiv preprint arXiv:2602.07764*, 2026.

Ryan Bahlous-Boldi, Isha Puri, Idan Shenfeld, Akarsh Kumar, Mehul Damani, Sebastian Risi, Omar Khattab, Zhang-Wei Hong, and Pulkit Agrawal. Vector policy optimization: Training for diversity improves test-time search. *arXiv preprint arXiv:2605.22817*, 2026. doi: 10.48550/arXiv.2605.22817.

Christopher M. Bishop. Mixture density networks. Technical Report NCRG/94/004, Aston University, 1994.

Ji Cheng, Xi Lin, Bo Xue, and Qingfu Zhang. Parametric pareto set learning: Amortizing multi-objective optimization with parameters. *IEEE Transactions on Evolutionary Computation*, 2026. doi: 10.1109/TEVC.2025.3630787.

Zelei Cheng, Xin-Qiang Cai, Yuting Tang, Pushi Zhang, Boming Yang, Masashi Sugiyama, and Xinyu Xing. UC-MOA: Utility-conditioned multi-objective alignment for distributional pareto-optimality. *arXiv preprint arXiv:2503.10669*, 2025.

Kalyanmoy Deb and Santosh Tiwari. Omni-optimizer: A generic evolutionary algorithm for single and multi-objective optimization. *European Journal of Operational Research*, 185(3):1062–1087, 2008. doi: 10.1016/j.ejor.2006.06.042.

Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.

Timo M. Deist, Monika Grewal, Frank J. W. M. Dankers, Tanja Alderliesten, and Peter A. N. Bosman. Multi-objective learning to predict pareto fronts using hypervolume maximization. *arXiv preprint arXiv:2102.04523*, 2021.

Abner Guzman-Rivera, Dhruv Batra, and Pushmeet Kohli. Multiple choice learning: Learning to produce multiple structured outputs. In *Advances in Neural Information Processing Systems*, volume 25, 2012.

Moksh Jain, Sharath Chandra Raparthy, Alex Hernández-García, Jarrid Rector-Brooks, Yoshua Bengio, Santiago Miret, and Emmanuel Bengio. Multi-objective GFlowNets. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 14631–14653. PMLR, 2023.

Yeong-Dae Kwon, Jinho Choo, Byoungjip Kim, Iljoo Yoon, Youngjune Gwon, and Seungjai Min. POMO: Policy optimization with multiple optima for reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 33, pp. 21188–21198. Curran Associates, Inc., 2020.

Stefan Lee, Senthil Purushwalkam Shiva Prakash, Michael Cogswell, Viresh Ranjan, David Crandall, and Dhruv Batra. Stochastic multiple choice learning for training diverse deep ensembles. In *Advances in Neural Information Processing Systems*, volume 29, pp. 2119–2127. Curran Associates, Inc., 2016.

Xi Lin, Zhiyuan Yang, and Qingfu Zhang. Pareto set learning for neural multi-objective combinatorial optimization. In *International Conference on Learning Representations*, 2022a.

Xi Lin, Zhiyuan Yang, Xiaoyuan Zhang, and Qingfu Zhang. Pareto set learning for expensive multiobjective optimization. In *Advances in Neural Information Processing Systems*, volume 35, pp. 19231–19247, 2022b. doi: 10.52202/068431-1398.

Xingchao Liu, Xin Tong, and Qiang Liu. Profiling pareto front with multi-objective stein variational gradient descent. In *Advances in Neural Information Processing Systems*, volume 34, pp. 14721–14733, 2021.

Aviv Navon, Aviv Shamsian, Ethan Fetaya, and Gal Chechik. Learning the pareto front with hypernetworks. In *International Conference on Learning Representations*, 2021.

Minh-Duc Nguyen, Phuong Mai Dinh, Quang-Huy Nguyen, Long P. Hoang, and Dung D. Le. Improving pareto set learning for expensive multi-objective optimization via stein variational hypernetworks. In *AAAI Conference on Artificial Intelligence*, 2025.

Biswajit Paria, Kirthevasan Kandasamy, and Barnabás Póczos. A flexible framework for multiobjective bayesian optimization using random scalarizations. In *Uncertainty in Artificial Intelligence*, pp. 766–776, 2020.

Yinuo Ren, Tesi Xiao, Michael Shavlovsky, Lexing Ying, and Holakou Rahmanian. COS-DPO: Conditioned one-shot multi-objective fine-tuning framework. In *Uncertainty in Artificial Intelligence*, 2025.

Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. In *Advances in Neural Information Processing Systems*, volume 28, 2015.

Yuki Suzumura, Yoshihiro Ohta, and Hiroyuki Sato. Handling multimodality in pareto set estimation via cluster-wise decomposition. *Applied Sciences*, 16(8):3655, 2026. doi: 10.3390/app16083655.

Ryoji Tanabe and Hisao Ishibuchi. A decomposition-based evolutionary algorithm for multimodal multi-objective optimization. In *Parallel Problem Solving from Nature – PPSN XV*, volume 11101 of *Lecture Notes in Computer Science*, pp. 249–261. Springer, 2018. doi: 10.1007/978-3-319-99253-2_20.

Ryoji Tanabe and Hisao Ishibuchi. A review of evolutionary multimodal multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 24(1):193–200, 2020. doi: 10.1109/TEVC.2019.2909744.

Tingyang Wei, Jiao Liu, Abhishek Gupta, Chin Chun Ooi, Puay Siew Tan, and Yew-Soon Ong. Amortized multi-objective optimization across tasks with generative solution modeling. In *International Joint Conference on Artificial Intelligence*, 2026.

Yucheng Yang, Tianyi Zhou, Mykola Pechenizkiy, and Meng Fang. Preference controllable reinforcement learning with advanced multi-objective optimization. In *International Conference on Machine Learning*, 2025.

Qingfu Zhang and Hui Li. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation*, 11(6):712–731, 2007.

Xinyu Zhang, Conor Hassan, Julien Martinelli, Daolang Huang, and Samuel Kaski. In-context multi-objective optimization. In *International Conference on Learning Representations*, 2026.

Eckart Zitzler, Kalyanmoy Deb, and Lothar Thiele. Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary Computation*, 8(2):173–195, 2000.

A Scope, Limitations, and Broader Impact

Serving assumptions. Candidate-set PASO requires a finite precommitted sequence whose repaired members can be evaluated accurately enough at query time. Expensive objectives restrict the usable candidate budget, while score noise can change the selected member. Appendix F.6 evaluates synthetic Gaussian selector noise on selected tasks, and Appendix F.7 reports retrospective feasibility-first re-scoring. These analyses characterize the evaluated tasks and perturbations rather than arbitrary noisy objectives or repair failures.

Metric scope. Quality-gated recall is defined over near-optimal equivalence classes discovered by the frozen multi-start teacher, not over a known complete solution set. Appendix F.1 specifies the teacher dependence, grouping, assignment, and quality gate, and Appendix F.3 analyzes the signed-permutation quotient used for the two neural-weight tasks. Other semantic equivalences may remain, so recall is interpreted under the stated teacher and task-specific equivalence rule.

Resource scope. The primary protocol matches stored slots and common final objective-scoring calls. Native output count, proposal-time objective calls, model evaluations, latency, parameter or archive storage, and offline computation remain source-dependent. Appendix B gives the serving-charge protocol, while Appendices F.2 and F.5 report archive and serving costs. The resulting comparisons are common-interface performance comparisons rather than equal end-to-end deployment-cost comparisons.

Theory and external validity. The squared-loss floor applies when every action in the complete constrained Bayes-action set has positive regret. The finite-k probability bounds require positive mass in the relevant post-repair regions and conditionally i.i.d. draws, and the selected-quality bound additionally assumes exact preference-specific scoring. Deterministic or correlated sequences retain the realized-hit implications but require a dependence model for probability guarantees; Appendix D gives the exact conditions. No candidate source dominates across all evaluated tasks, and behavior with more objectives, more solution regions, or different semantic equivalences remains outside the current evaluation.

Broader impact. Exposing several scored alternatives can make trade-offs and omitted solution regions visible, but the serving interface does not determine which trade-off should be chosen and inherits the objectives, data, reference construction, and equivalence rules used to generate and evaluate candidates. Because preference-conditioned serving fixes a balance among competing costs, we report per-preference selected quality, feasibility, and teacher-discovered quality-gated recall alongside aggregate front quality. The Adult-specific data and deployment boundaries remain in the Ethics Statement.

B Supervision and Serving-Charge Protocol

This appendix states which teacher object supervises each source and how many objective evaluations each source uses at serving time (Table 3).

Two teacher objects. For a fixed preference, the *near-endpoint law* is the empirical law on the outputs of the $S{=}32$ random-start, $K{=}100$-step teacher whose scalarized values pass the near-optimal tolerance of Appendix F.1; it contains at most 32 endpoints. The *mode-representative law* is the empirical law on the greedy decision-space cluster representatives of that near set. These are different training objects: Endpoint, MCL, MDN, and cVAE are fitted to the former under L_2 regression, winner-takes-all L_2, mixture NLL, and the ELBO respectively, whereas PCGRF is fitted to the representatives by flow matching and the archive simply stores them. PHN-obj and its ensemble and multi-head variants use no teacher at all: they are trained directly on the scalarized objective, which is exactly the supervision contrast that Section 5.2 exploits. Archive retrieval and random search have no fitted loss. The pooled archive retains every discovered representative, so its size is task- and seed-dependent, with at most $16 \times 32 = 512$ entries for the 16 training preferences (Appendix F.2). The representative set also supplies the per-dimension scale used by PCGRF and by the point-plus-noise sources; the point-plus-noise comparator fixes its multiplier at 1.

Serving-time charges. Every source in the comparison precommits its sequence and incurs only the 64 common scoring evaluations per query (Table 3). Network evaluations are counted separately: a point map or a multi-head map uses one model evaluation, an eight-member ensemble uses eight, and MDN and cVAE use one batched conditional evaluation rather than 64 independent forward passes. A point row scores its single decision in all 64 slots, so its timing is not native one-point deployment latency.

Table 3: Per-query slot accounting at k=64. "Native" is the number of candidates a source produces before repetition fills the stored slots; A is the archive size. Every source makes no proposal-time objective call and receives exactly 64 common scoring calls, which exclude the offline teacher and the reference construction.

Policy	Native output	Expansion to 64 stored slots
Endpoint / aligned Endpoint; PHN-obj	1 point	duplicate the point 64 times
Endpoint / PHN-obj + noise	64 draws	base in slot 1 plus 63 Gaussian perturbations
Endpoint / PHN-obj ens.-8; aligned ens.-8	8 member outputs	cyclic tile; each member occupies 8 slots
MCL-8 / MCL-8+rep.; PHN-obj mh.-8+rep.	8 head outputs	cyclic tile; each head occupies 8 slots
MDN / MDN-16	64 mixture draws	sample 64 directly ($C = 8/16$ is not an output count)
cVAE	64 latent draws	sample 64 directly
PCGRF	64 source trajectories	sample and transport 64 directly
Archive (nearest, interpolated)	1 row	broadcast the retrieved or interpolated row 64 times
Archive (k-nearest, global)	$\min(64, A)$ rows	truncate at 64; tile only when $A < 64$
Random search	64 draws	sample 64 directly

C THE SERVING ALGORITHM

Algorithm 1 writes out the common precommit-then-score procedure of Section 4.1 as pseudocode; every candidate source in the serving study is served by exactly this loop, and only the first line differs between sources. The offline multi-restart teacher that supervises the imitation-trained sources is a separate construction (Appendix F.1).

Algorithm 1 Candidate-set PASO serving

Require: Frozen source Gen_θ, preference $\boldsymbol{\lambda}$, stored budget k, optional randomness ξ, repair $\mathsf{Rep}_{\mathcal{X}}$, objectives $\boldsymbol{F}$, scalarization $s_{\boldsymbol{\lambda}}$

1: Generate the complete $\mathbf{Z}_\theta^k(\boldsymbol{\lambda}, \xi)$ before common final scoring
2: **for** $j = 1, \ldots, k$ **do**
3: $\quad \boldsymbol{x}^{(j)} \leftarrow \mathsf{Rep}_{\mathcal{X}}(\boldsymbol{z}^{(j)})$
4: $\quad v_j \leftarrow s_{\boldsymbol{\lambda}}(\boldsymbol{F}(\boldsymbol{x}^{(j)}))$
5: **end for**
6: $\widehat{j} \leftarrow \min \arg\min_{j \in [k]} v_j$
7: Let $\mathcal{I}$ contain the first index of each distinct repaired decision
8: **return** $\boldsymbol{x}^{(\widehat{j})}$ and, if requested, $\{(\boldsymbol{x}^{(j)}, v_j) : j \in \mathcal{I}\}$

How each source family instantiates line 1. Lines 2–8 are shared verbatim by every policy in the serving comparison, so the whole methodological difference between families is which sequence line 1 emits. A point map evaluates its network once and the expansion rule T_k repeats that single decision into all k slots, which is why a point policy still pays the full k-call scoring charge in our protocol: it buys no diversity with it. A deterministic set evaluates J members or heads once each and tiles them cyclically, so the realized distinct count is at most J regardless of k. A conditional generator draws k independent latents and decodes them in one batched forward, which is the only family whose realized distinct count grows with the budget. Archive retrieval performs no network evaluation at all: it re-scalarizes the cached objective vectors of all N stored entries under $\boldsymbol{\lambda}$, sorts them, and returns the best k: an online cost that grows with the store rather than with the model. The precommitment requirement is what makes these comparable, because it forbids any of them from using a score returned in line 5 to choose a later candidate.

D Proofs for the Main Mechanism Results

Lemma 1 (constrained Bayes actions). For every $\boldsymbol{a} \in \mathbb{R}^d$,

$$\mathbb{E}\|\boldsymbol{a} - \boldsymbol{Y}\|_2^2 = \|\boldsymbol{a} - \boldsymbol{m}_\mu\|_2^2 + \mathbb{E}\|\boldsymbol{Y} - \boldsymbol{m}_\mu\|_2^2. \tag{4}$$

The second term is independent of $\boldsymbol{a}$, so minimization over $\mathcal{X}$ gives exactly the Euclidean projection set $\mathrm{Proj}_{\mathcal{X}}(\boldsymbol{m}_\mu)$. Continuity of the squared distance and compactness of $\mathcal{X}$ make this set nonempty; nonconvexity can make it non-singleton. The definition of Δ_μ then gives the displayed regret bound uniformly over all Bayes actions. For the symmetric two-point law the mean is the midpoint; when it is feasible, its distance to itself is zero and the projection set is the singleton containing it. A positive floor additionally requires more than lying outside selected mode neighborhoods. One sufficient route is that the neighborhoods contain the entire zero-regret set, the complete projection set $\mathcal{A}_\mu$ lies in their complement, and

$$\inf_{\boldsymbol{x} \in \mathcal{X} \setminus \cup_j B_\rho(\boldsymbol{x}_j^\star)} [g(\boldsymbol{x}) - g^\star] > 0. \tag{5}$$

Lemma 2 (single-candidate recall). Because the membership cells D_j are pairwise disjoint, at most one membership indicator can be one for any realized candidate. The quality gate can only turn such an indicator off, so $\sum_j I_j^{\varepsilon_{\mathrm{th}}}(\boldsymbol{x}) \le 1$ and division by M proves the bound. The argument is independent of training loss and holds for every realization, hence also in expectation. Nearest assignment labels a candidate but does not guarantee that it passes any quality gate; the upper bound may therefore be strict, including zero.

Theorem 1 (candidate-set bounds). Let E be the event that at least one of the k draws lies in $\cup_j B_j$. Conditional independence gives $\Pr(E^c) = (1 - q_\cup)^k$. On E, exact selection returns a candidate with regret at most b_ρ; on E^c, its regret is at most $R_{\max}$. Thus

$$\mathbb{E}[g(\widehat{\boldsymbol{X}}_k) - g^\star] \le b_\rho + (1 - q_\cup)^k R_{\max} \le b_\rho + \left(1 - \sum_j p_j\right)^k R_{\max}. \tag{6}$$

For each mode j, the miss probability is $(1 - q_j)^k \le (1 - p_j)^k$. A union bound, clamped at zero, gives the all-mode statement. Linearity of expectation applied to the mode-hit indicators gives $\mathbb{E}[\mathrm{Rec}_{\varepsilon_{\mathrm{th}}}] \ge M^{-1} \sum_j [1 - (1 - p_j)^k]$; independence between the different mode-hit events is unnecessary. The Lipschitz choice $b_\rho = G\rho$ is immediate. The quadratic choice follows from the descent lemma when each ball lies in a convex neighborhood where the mode is stationary and the gradient is L-Lipschitz; this condition requires stationary modes rather than boundary optima.

The probability bounds apply to conditionally i.i.d. draws from the final post-repair law. For deterministic, ensemble, retrieval, or otherwise correlated sequences, the analysis gives realized-hit implications: a realized member in B_j certifies regret at most b_ρ, and realized hits in all M neighborhoods certify full recall when $b_\rho \le \varepsilon_{\mathrm{th}}$. Finite-$k$ probability guarantees for these sequences require an additional dependence model.

Instantiation on Bimodal2D. Let $h(x_1, x_2) = \min\{(x_1 + 1)^2 + x_2^2, (x_1 - 1)^2 + x_2^2\}$ and $F_1 = F_2 = h$. With anchors $(0, 0)$ and $(4, 4)$, the equal preference remains $(1/2, 1/2)$ after smoothing, so

$$g(\boldsymbol{x}) = \tau \log 2 + \frac{h(\boldsymbol{x})}{8 + 2\delta_{\mathrm{norm}}}. \tag{7}$$

At the modes $\boldsymbol{x}_a = (-1, 0)$ and $\boldsymbol{x}_b = (1, 0)$, $h = 0$; at their midpoint $\boldsymbol{m} = (0, 0)$, $h = 1$. The exact midpoint-to-mode gap is therefore $1/(8 + 2\delta_{\mathrm{norm}}) = 0.1249999996875\ldots$, or $1/8 = 0.125$ when the numerical stabilizer is omitted, and is independent of τ. Along $x_2 = 0$, the left and right derivatives of h at zero are 2 and -2. Thus the midpoint is a nondifferentiable cusp and a strict local maximum along that line, not a smooth stationary point. The two modes are interior stationary minima with a unique quadratic branch locally, so the local quadratic candidate bound applies there. The learned Endpoint frozen-reference gap 0.1145 ± 0.0061 and the teacher-restart centroid diagnostic 0.0929 are empirical quantities distinct from the analytic gap.

Theorem 2 (endpoint and separator obstruction). Compactness and joint continuity imply continuity of $v(s) = \min_{\boldsymbol{x}\in\mathcal{X}} g_{\boldsymbol{\lambda}(s)}(\boldsymbol{x})$, and hence of $s \mapsto r(s, h(s))$. If $h(a) \notin N_A$ or $h(b) \notin N_B$, an endpoint has regret at least γ. Otherwise the connected image $h([a, b])$ meets both disjoint closed sets. It cannot be contained in their union, since their intersections with the image would then disconnect it. Thus $h(s_*) \notin N_A \cup N_B$ for some $s_* \in (a, b)$, where the separator assumption gives regret at least γ. In either case continuity yields a nonempty open subinterval of (a, b) with regret at least $\gamma/2$. The interval may depend on h; a query density positive almost everywhere assigns it positive probability but supplies no uniform probability lower bound.

A continuous example with crossing branch values. Take $\mathcal{X} = [-1, 1]$, $s \in [-1/8, 1/8]$, and $f(s, x) = (1 - x^2)^2 - sx$. Its minimum is $-|s|$, attained uniquely at -1 for $s < 0$ and 1 for $s > 0$, with both endpoints optimal at zero. Set $N_A = [-1, -1/2]$ and $N_B = [1/2, 1]$. Outside their union, $f(s, x) \geq 9/16 - 1/16 = 1/2$, so regret is at least $1/2$. At $s = -1/8$, points $x \geq 0$ have $f(s, x) \geq 0$ and regret at least $1/8$; points $-1/2 < x < 0$ have still larger regret. The other endpoint follows by symmetry, verifying the theorem with $\gamma = 1/8$. Meanwhile the wrong endpoint's regret is $2|s|$, which tends to zero at the crossing. The theorem therefore requires no positive gap between competing branches near their crossing.

ZDT3 under the experimental scalarizer. Writing $u = x_1$ and $G = 1 + 9\sum_{i=2}^{30} x_i/29$, the second objective is $f_2 = G - \sqrt{uG} - u\sin(10\pi u)$. Since $\partial f_2/\partial G = 1 - \frac{1}{2}\sqrt{u/G} \geq 1/2$ and the smoothed scalarizer is strictly increasing in each objective, all global minimizers have $x_2 = \cdots = x_{30} = 0$. This also reduces restricted minima on sets defined by x_1 to one dimension. We use the experimental ideal $(0, -1)$, nadir $(3.5, 3.5)$, normalization offset 10^{-8}, temperature 0.05, and weight smoothing 10^{-4}. A scan and local branch-value root refinement locate switches at $\lambda_1 \approx 0.394835, 0.527309, 0.658289, 0.785239$. For the first, use $[a, b] = [0.3928347653, 0.3968347653]$, $N_A = \{\boldsymbol{x} : x_1 \geq 0.7697115805\}$, and $N_B = \{\boldsymbol{x} : x_1 \leq 0.7297115805\}$. The competing minimizers at the crossing have $x_1 \approx 0.84820230$ and 0.65119981.

We check margins on a uniform grid in $t = \sqrt{u}$ with step 10^{-5} and 401 equally spaced preferences on each interval. The scalarizer has Lipschitz bounds $L_t \leq (3 + 20\pi)/4.5$ and $L_s \leq 2/3$. Subtracting $L_t 10^{-5}$ from restricted grid minima and additionally $2L_s 10^{-5}$ for the separator's preference scan controls discretization error. A sampled feasible reference gives an upper bound on the true optimal value. After these corrections and a 10^{-10} numerical slack, the first interval has endpoint margin at least 1.438×10^{-4} and separator margin at least 0.12657; the other three minimum margins exceed 1.74×10^{-4}. These computations support $\gamma = 10^{-4}$ on all four intervals using ordinary double precision, rather than interval arithmetic. They check the theorem's geometry; they do not identify the source of a trained model's aggregate gap or restrict a discontinuous retrieval policy.

Remark 1 (Asymmetric and multimodal teacher laws)**.** *For any teacher law with mean $\boldsymbol{m}_\mu$, including a weighted finite-mode law, the constrained Bayes-action set is $\mathrm{Proj}_{\mathcal{X}}(\boldsymbol{m}_\mu)$ and its guaranteed floor is $\inf_{\boldsymbol{a}\in\mathrm{Proj}_{\mathcal{X}}(\boldsymbol{m}_\mu)}[g(\boldsymbol{a}) - g^\star]$. Neither nonuniform weights nor location outside selected mode neighborhoods alone guarantees positivity, and the floor can vanish before the law concentrates on one named mode if another zero-regret action lies at a projected centroid. The empirical mean-collapse diagnostic evaluates the restart centroid with its observed weights and provides instance-specific evidence. With M disjoint quality-gated cells, one candidate recalls at most one mode and can recall none.*

Remark 2 (Symmetry-induced multiplicity and the quotient)**.** *For the one-hidden-layer tanh decision networks used in the digit-pair and Adult-MLP tasks, permuting hidden units and coordinating sign flips of their incoming weights, biases, and outgoing weights preserves the represented function. These orbit elements are representation-equivalent rather than user-distinct alternatives. They can contribute to squared-loss averaging, but Lemmas 1–2 and Theorem 1 do not apply to a group orbit verbatim: the orbit law may be nonuniform, its centroid need not have positive regret, and general symmetry groups need not be finite. We therefore canonicalize the known signed-permutation orbit before regression and report recall over the finite teacher-discovered classes that remain in the quotient. Residual classes enter these results only when they are decision-relevant and satisfy the remaining assumptions; the quotient recomputation does not independently validate that status. This distinction concerns the tanh task networks; the SiLU amortizer networks are outside the sign-flip symmetry analysis.*

E SUPPLEMENTARY DIAGNOSTICS

This appendix supplies what Section 5 could not fit: how the frozen references are built, the task and amortizer specifications, the complete single-candidate-versus-candidate-set matrix behind Table 2, and two checks on the reference and on a second real-data task pair.

E.1 FROZEN-REFERENCE CONSTRUCTION

Every serving study uses a task- and seed-specific frozen reference. For the multimodal and data-derived tasks this is the union of representatives found by an evaluation teacher on 64 frozen preferences, with $S{=}32$ restarts and $K{=}100$ projected steps, followed by the near-optimality filter and clustering of Appendix F.1 at 0.03 of the box diagonal. The remaining tasks use a policy-independent analytic or seeded base construction followed by nondominance filtering. In both cases the reference is fixed before candidate scoring, and evaluated-policy outputs are never admitted to its construction.

E.2 TASK SPECIFICATIONS AND AMORTIZER HYPERPARAMETERS

Tasks. Decision dimensions and objective counts are: Bimodal2D ($d{=}2$, $m{=}2$), OmniTest $(2, 2)$, ZDT3 $(30, 2)$, RE24 $(2, 2)$, digit-pair MTL $(2404, 2)$, Adult-fair linear $(43, 2)$, and Adult-fair MLP $(705, 2)$; all tasks use box-constrained decision spaces. Digit-pair MTL concatenates two scikit-learn digits images into a 128-dimensional input; the decision variable is the flattened weight vector of a shared-trunk tanh MLP (hidden width 16) with two linear classification heads, bounded to $[-1.5, 1.5]$; objectives are the two cross-entropy losses on a fixed 1024-pair subset, normalized by chance level $\ln 10$. The Adult-fair pair uses a fixed 2048-row subset of the Adult census data (42 features after standardization and one-hot encoding; sex excluded from inputs and used only as the group mask): objectives are income cross-entropy and a smooth demographic-parity gap, with upper scaling anchor $(\ln 2, 0.5)$; the decision variable is the weights of a logistic model ($d{=}43$) or a width-16 tanh MLP ($d{=}705$), bounded to $[-1.5, 1.5]$; held-out evaluation splits for both data-derived tasks are defined in Appendix F.4. Table 4 lists the objective-normalization anchors. Some coincide with analytic Pareto-front ideal and nadir points, whereas others are benchmark constants or task-defined scales; all are resolved from the task configuration before preferences, teacher outputs, or policy outputs are constructed.

Table 4: Fixed lower/upper objective-normalization anchors. Anchors are resolved from the task configuration before preferences, teacher solutions, or policy outputs are constructed and are shared across methods and seeds; upper anchors are reference scales rather than Pareto-front nadirs unless stated. The RE24-C row is used only for the constrained re-scoring of Appendix F.7.

Task	Lower anchor $\boldsymbol{a}$	Upper anchor $\boldsymbol{b}$	Source / interpretation
Bimodal2D	$(0, 0)$	$(4, 4)$	Custom task scale; $\boldsymbol{b}$ is neither the PF nadir nor a global objective bound.
OmniTest	$(-2, -2)$	$(2, 2)$	Analytic per-objective coordinate range, matching the benchmark constants.
ZDT3	$(0, -1)$	$(3.5, 3.5)$	Fixed padded reference box, not exact PF endpoints.
RE24	$(60.5, 0)$	$(481.608088535,$ $44.2819047619)$	Fixed constants of the implemented RE24 formulation.
Digit-pair	$(0, 0)$	$(\ln 10, \ln 10)$	Task-defined chance-classification-loss scale.
Adult linear	$(0, 0)$	$(\ln 2, 0.5)$	Chance-level BCE plus a task-defined parity-gap scale; neither upper value is a global bound.
Adult MLP	$(0, 0)$	$(\ln 2, 0.5)$	Same task-defined scales as Adult linear.
RE24-C re-score	$(484.8231201171875,$ $0)$	$(5950.984375,\ 1)$	Deterministic $801{\times}801$ feasible-design grid; the upper f_1 anchor is its 99th percentile, not a nadir.

Learned sources. The default MDN is a deliberately fixed low-capacity configuration used unchanged on every task: $C{=}8$ Gaussian components with a minimum scale of 10^{-3}. MDN-16 doubles both capacity knobs ($C{=}16$, hidden width 256) with all other settings identical; the cVAE uses an 8-dimensional latent with KL weight 10^{-3}; PCGRF uses a center-Gaussian source and 16 integration steps. All networks except MDN-16 use hidden width 128, and all are trained for 600 epochs with Adam at learning rate 10^{-3}.

Table 5: Complete serving matrix on Bimodal2D and OmniTest under 64 stored slots and 64 common scoring calls (means over ten seeds; $\pm$ is the seed standard deviation). Best-of-k gain is the mean candidate gap minus the selected candidate's gap; "Cover" is raw nearest-assignment coverage without the quality gate of Table 2. Bold marks the best gap and the best coverage per task; small negative gaps reflect approximate-reference slack.

Task	Method	Ref. gap ↓	Best-of-k gain	Cover ↑
Bimodal2D	Endpoint	0.1145 ± 0.0061	0.0000	0.500
	PHN-obj	**0.0000 ± 0.0000**	0.0000	0.500
	Endpoint+noise (σ=1)	6.5e-05 ± 1.6e-05	0.0460	**1.000**
	PHN-obj+noise (σ=1)	**0.0000 ± 0.0000**	0.0520	**1.000**
	Endpoint ens.-8	0.1086 ± 0.0082	0.0075	0.601
	PHN-obj ens.-8	**0.0000 ± 0.0000**	0.0000	**1.000**
	MCL-8	**0.0000 ± 0.0000**	0.0773	**1.000**
	MCL-8+rep.	1.6e-05 ± 1.1e-05	0.4414	**1.000**
	PHN-obj mh.-8+rep.	**0.0000 ± 0.0000**	0.4811	**1.000**
	Random search	0.0048 ± 4.6e-04	0.2039	**1.000**
	MDN	1.2e-04 ± 4.3e-05	0.0466	**1.000**
	MDN-16 (high cap.)	2.9e-04 ± 4.8e-04	0.0443	**1.000**
	cVAE	4.6e-05 ± 2.0e-05	0.0426	**1.000**
	PCGRF (σ=1)	1.7e-06 ± 1.8e-06	0.0025	**1.000**
OmniTest	Endpoint	0.0715 ± 0.0423	0.0000	0.118
	PHN-obj	3.8e-04 ± 1.8e-05	0.0000	0.118
	Endpoint+noise (σ=1)	0.0107 ± 0.0032	0.3093	0.995
	PHN-obj+noise (σ=1)	3.7e-04 ± 1.9e-05	0.3179	0.996
	Endpoint ens.-8	0.0339 ± 0.0273	0.0296	0.123
	PHN-obj ens.-8	**1.9e-04 ± 1.9e-05**	2.0e-04	0.124
	MCL-8	2.9e-04 ± 2.2e-04	0.0566	0.848
	MCL-8+rep.	0.0016 ± 9.5e-04	0.0895	0.856
	PHN-obj mh.-8+rep.	0.0010 ± 5.7e-04	0.3586	0.869
	Random search	0.0136 ± 0.0016	0.3014	**0.999**
	MDN	9.1e-04 ± 6.1e-04	0.1689	0.998
	MDN-16 (high cap.)	0.0042 ± 0.0056	0.2009	0.994
	cVAE	0.0133 ± 0.0019	0.3025	0.713
	PCGRF (σ=1)	0.0041 ± 0.0012	0.2517	0.997

Perturbation scale. The teacher-derived scale shared by PCGRF and the point-plus-noise sources is the per-dimension standard deviation of the teacher's restart mode representatives, floored at 10^{-3} of the box width. The point-plus-noise comparator uses a fixed construction multiplier $\sigma_{\mathrm{con}} = 1$ that was not tuned on test preferences.

Diversity-trained multi-head sources. MCL-8+rep. and PHN-obj mh.-8+rep. share the MCL architecture (8 heads, hidden width 128) and add a mutual-repulsion term, the mean RBF kernel $\exp(-\overline{\|\boldsymbol{z}_i - \boldsymbol{z}_j\|^2}/h^2)$ over head pairs in the box-normalized decision space (h=0.2, weight 1.0). MCL-8+rep. trains winner-takes-all imitation plus repulsion; PHN-obj mh.-8+rep. trains the *best head's* scalarized objective plus repulsion, so non-winning heads receive a diversity gradient instead of collapsing onto the winner, which makes it a budget-matched representative of Stein-variational-style diversity promotion (Nguyen et al., 2025).

Aligned estimator and archive policies. The symmetry-aligned estimator of Appendix F.3 shares every hyperparameter with endpoint regression and differs only in that its targets are canonicalized first. The archive policies of Appendix F.2 introduce no fitted hyperparameters: they reuse the same teacher and preference grid and enter the same 64-slot protocol, although their native output count, archive size, sorting work, and storage are not matched to the neural policies.

E.3 Full single-candidate-versus-candidate-set matrix

Tables 5 and 6 expand Table 2 to every source on the four headline tasks, with standard deviations, best-of-k gains, and raw coverage; Table 7 gives the same comparison on the Adult pair. Every row uses 64 stored selector slots and 64 final objective-scoring calls, not 64 network forwards: point and multi-head maps use one model evaluation, ensembles use eight, and generators use one batched conditional evaluation. Figure 3 reports the seed-level uncertainty of the quality-gated recall behind Table 2.

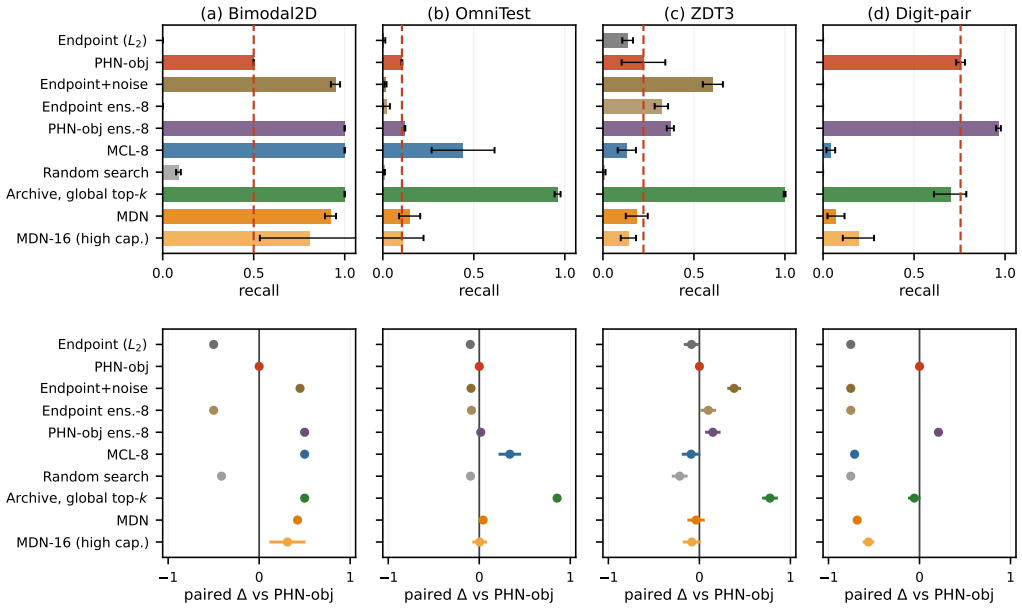


Figure 3: Quality-gated recall uncertainty over seeds 0–9 for every policy in the serving comparison. *Top:* recall means with sample-SD error bars; the dashed line marks PHN-obj, the strongest learned single-candidate policy. *Bottom:* seed-paired differences relative to PHN-obj with two-sided 95% Student-t intervals over the ten paired differences (df=9), no multiplicity correction; a point to the right of zero favours the row. Bimodal2D, OmniTest, and ZDT3 use raw nearest assignment, digit-pair the signed-permutation quotient, and Endpoint+noise the fixed $\sigma_{\text{con}} = 1$ comparator. Most paired intervals are narrower than the plotted markers.

E.4 Is the separation an artifact of the reference or of the task pair?

Two checks ask whether the reported separation survives a change of yardstick and a change of data.

The first re-scores stored digit-pair outputs against a much stronger oracle to estimate the frozen reference's slack. Figure 4 shows a common shift of about 0.04, which leaves the objective-trained maps at roughly -0.006, as an approximate reference predicts. Because every policy is scored against the same reference at each query, pairwise gap differences, and hence the ordering, are algebraically invariant to the change of reference; the preserved ordering is therefore a consistency check, not evidence that the reference is reliable. The check re-scores stored best values without reconstructing reference modes, so it says nothing about recall under a stronger teacher.

The second uses the Adult fairness–accuracy pair, a second real-data task pair that was fixed before any serving method was run on it. Table 7 reports it: the linear task ties near zero for every family, whereas the MLP task reproduces the squared-loss collapse. The first check calibrates the scale of the reported gaps and the second shows that the separation is not specific to one benchmark pair; neither establishes that the discovered alternatives are complete.

F Retrieval, Symmetry, Cost, Constraints, and Robustness

This appendix expands the analyses the main text summarizes. Sections F.1–F.5 define how modes and recall are measured, evaluate non-parametric retrieval from the offline teacher archive and symmetry-aware point estimation on the weight-space tasks, report the held-out-data evaluation, and decompose per-query serving cost. Section F.6 asks whether the conclusions survive an imperfect selector, and Section F.7 closes with a constrained re-scoring of RE24, where a query can fail outright.

Table 6: Complete serving matrix on ZDT3 and digit-pair MTL; columns and conventions as in Table 5.

Task	Method	Ref. gap ↓	Best-of-k gain	Cover ↑
ZDT3	Endpoint	0.0495 ± 0.0063	0.0000	0.947
	PHN-obj	0.0369 ± 0.0034	0.0000	0.947
	Endpoint+noise (σ=1)	0.0069 ± 9.3e-04	0.0448	0.995
	PHN-obj+noise (σ=1)	0.0346 ± 0.0024	0.0088	0.952
	Endpoint ens.-8	0.0241 ± 0.0085	0.0254	0.978
	PHN-obj ens.-8	0.0339 ± 2.3e-04	0.0020	0.952
	MCL-8	0.0476 ± 0.0128	0.3699	0.997
	MCL-8+rep.	0.0694 ± 0.0170	0.4179	**1.000**
	PHN-obj mh.-8+rep.	0.0404 ± 8.7e-04	0.3801	**1.000**
	Random search	0.2786 ± 0.0040	0.2009	**1.000**
	MDN	0.0073 ± 0.0017	0.1029	**1.000**
	MDN-16 (high cap.)	**0.0062 ± 0.0015**	0.0932	**1.000**
	cVAE	0.0341 ± 0.0051	0.0515	0.995
	PCGRF (σ=1)	0.0095 ± 0.0049	0.0464	0.986
Digit-pair MTL	Endpoint	0.5705 ± 0.0504	0.0000	0.756
	PHN-obj	-0.0447 ± 0.0037	0.0000	0.756
	Endpoint+noise (σ=1)	0.5047 ± 0.0482	0.2150	0.882
	PHN-obj+noise (σ=1)	-0.0447 ± 0.0037	0.0995	0.815
	Endpoint ens.-8	0.4879 ± 0.0409	0.0813	0.802
	PHN-obj ens.-8	**-0.0453 ± 0.0036**	7.8e-04	0.990
	MCL-8	0.0970 ± 0.0246	0.6823	0.981
	MCL-8+rep.	0.1319 ± 0.0234	0.8067	0.986
	PHN-obj mh.-8+rep.	-0.0427 ± 0.0041	2.4600	0.992
	Random search	1.2335 ± 0.0084	0.4813	**1.000**
	MDN	0.1224 ± 0.0697	0.7323	0.996
	MDN-16 (high cap.)	0.0266 ± 0.0102	0.4475	0.995
	cVAE	0.3357 ± 0.0353	0.3539	0.996
	PCGRF (σ=1)	0.2808 ± 0.0776	0.2243	0.890

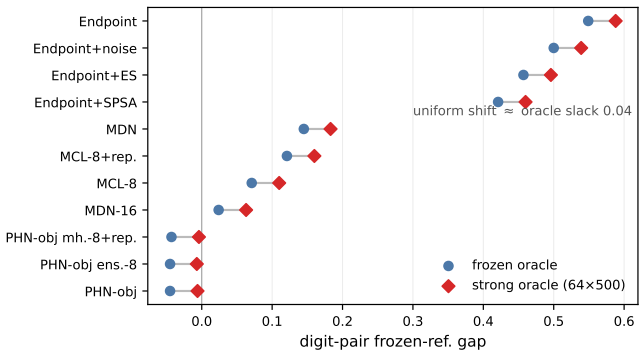


Figure 4: Strong-teacher check on digit-pair (seeds 0–2, 16-preference subset; means over seeds, with the three construction-noise multipliers of the point-plus-noise source pooled). Re-scoring every stored policy against a 64-restart, K=500 oracle shifts all frozen-reference gaps by the reference's slack (≈0.04). Pairwise differences are algebraically invariant to this shared shift, so the preserved ordering calibrates the gap scale and does not by itself validate the reference or its mode set. Endpoint+ES and Endpoint+SPSA are local-search refinements of the point map (an evolution strategy and SPSA started from its output) that are not part of the precommitted comparison.

F.1 How modes and recall are defined

Every mode-based quantity in the paper is derived from the same frozen teacher, by the following procedure, applied per preference. The S=32 restarts of the K=100-step projected scalarized solver produce outputs with scalarized values $\{g_i\}$; write $g^{\min} = \min_i g_i$ and let the *near-optimal* set be $\{i : g_i \le g^{\min} + \eta\}$ with $\eta = \max(10^{-5}, 0.05\,(q_{90}(g) - g^{\min}))$, i.e. a relative tolerance on the restart value spread with an absolute floor. The near-optimal outputs are then clustered greedily in *unnormalized* decision space, in order of increasing scalarized value: each point either joins an existing cluster whose representative lies within ρ, or starts a new one, where $\rho = 0.03\,\|\boldsymbol{x}^u - \boldsymbol{x}^l\|_2$ is three percent of the box diagonal. The cluster representatives $\{\widetilde{\boldsymbol{x}}_j^\star\}$ are the task's *discovered* modes

Table 7: The Adult fairness–accuracy pair (same protocol and symbols as Table 2). Every family ties near zero on the linear task, whereas the MLP task reproduces the squared-loss Endpoint collapse. PHN-obj's small negative frozen-reference gap is approximate-reference slack.

		Adult-fair (linear)			Adult-fair (MLP)		
Family	Method	Ref. gap ↓	±	Recall ↑	Ref. gap ↓	±	Q-recall ↑
Point map	Endpoint	0.0024	0.0010	0.314	0.2190	0.0274	9.8e-04
	PHN-obj	5.9e-04	3.0e-04	0.403	-0.0125	0.0027	0.797
Constructed dist.	Endpoint+noise	8.7e-04	2.2e-04	0.372	0.1574	0.0120	0.013
	PHN-obj+noise	4.1e-04	1.2e-04	0.436	-0.0129	0.0027	0.831
	Endpoint ens.-8	0.0011	4.3e-04	0.420	0.1820	0.0239	0.006
	PHN-obj ens.-8	**1.7e-04**	5.3e-05	**0.624**	**-0.0165**	0.0011	**0.950**
	MCL-8	0.0011	2.4e-04	0.377	0.0649	0.0227	0.059
	Random search	0.1964	0.0045	9.5e-04	0.2988	0.0032	1.1e-04
Learned dist.	MDN	0.0037	9.6e-04	0.066	0.0632	0.0296	0.077
	MDN-16 (high cap.)	0.0026	1.8e-04	0.094	0.0242	0.0158	0.186
	cVAE	0.0012	5.0e-04	0.384	0.1261	0.0070	0.010
	PCGRF	0.0046	7.8e-04	0.023	0.0816	0.0155	0.009

at that preference, and the fraction of preferences admitting more than one is that task's multimodal fraction.

This construction has two consequences. First, these are the modes the restarts *found*: the metric is properly a discovered-mode recall and it inherits the teacher's exploration bias, so we never treat the mode set as ground truth. Second, for the two neural-weight tasks we quotient the known hidden-unit signed-permutation action before evaluating recall. Teacher representatives whose quotient distance is numerically zero are placed in one class; a served candidate is assigned to the nearest class by Hungarian alignment, and the quality threshold is relative to the best teacher member of that class. In practice no two teacher representatives fall in the same orbit, but quotient assignment still changes the nearest class for some candidates. Other tasks use raw decision-space nearest assignment. A served candidate set recalls class j at level $\varepsilon_{\mathrm{rec}}$ when some member is assigned to class j by that task-appropriate nearest assignment *and* has scalarized value within $\varepsilon_{\mathrm{rec}}$ of the class's best teacher representative. The metric uses two design choices. The quality threshold is relative to the mode rather than to the global optimum, because the representatives sit inside the teacher's near-optimal tolerance and an absolute threshold would grade the teacher rather than the serving policy. And membership is by assignment rather than by a fixed radius: a radius that is a constant fraction of the box diagonal stops meaning anything once d is in the hundreds, where every candidate is far from every mode in L_2 and the radius version reads zero for every policy. We use $\varepsilon_{\mathrm{rec}} = 10^{-3}$ throughout.

F.2 Offline-archive retrieval

Question. Every imitation-trained amortizer is fitted to teacher endpoints that were computed offline, so the obvious challenge to any of them is: why not simply keep those endpoints and look them up? This section answers it under the same protocol the main text uses.

Setup. Four retrieval policies enter the same 64-slot protocol as Table 2. *Nearest* returns the lowest-scalarized endpoint of the nearest training preference and *interpolated* the distance-weighted decision-space interpolation of the two nearest, so both are single-candidate policies; *k-nearest set* returns the union of the nearest training preferences' mode representatives and *global top-k* ranks the whole pooled archive by its precomputed scalarized value and returns the best k, so both are candidate sets. The archive holds only *training*-preference teacher outputs, so no test-preference information can leak into it. Its objective vectors are precomputed, so ranking costs no new objective calls, but each query still re-scalarizes and fully sorts all A stored entries: ranking work and memory grow with the task- and seed-dependent A, not with a fixed parameter count.

Result. Table 8 reports the results. Two patterns matter. First, decision-space interpolation across neighbouring preferences behaves as a non-parametric analogue of squared-loss mean collapse: it lands between branches wherever the correspondence is set-valued or disconnected, and it is the worst retrieval policy on every task but Bimodal2D. That the same failure appears with no network

Table 8: Offline-archive retrieval under the same 64-slot protocol as Table 2. The archive retains every discovered representative, so its size is task- and seed-dependent. Recall is quality-gated discovered-mode recall ($\varepsilon_{\text{rec}} = 10^{-3}$); the two neural-weight columns use signed-permutation-quotient assignment (Q-recall). Bold marks the best value per column.

	Bimodal2D		OmniTest		ZDT3		Digit-pair		Adult-fair (MLP)	
Policy	Ref. gap ↓	Recall ↑	Ref. gap ↓	Recall ↑	Ref. gap ↓	Recall ↑	Ref. gap ↓	Q-recall ↑	Ref. gap ↓	Q-recall ↑
Endpoint (L_2)	0.1145	0.000	0.0715	0.006	0.0495	0.136	0.5705	0.000	0.2190	9.8e-4
PHN-obj	**-4.5e-10**	0.500	3.8e-4	0.105	0.0369	0.223	**-0.0447**	**0.756**	**-0.0125**	0.797
Archive, nearest $\boldsymbol{\lambda}$	2.7e-8	0.500	1.9e-4	0.118	2.4e-4	0.903	0.0156	0.357	0.0103	0.466
Archive, interpolated	1.6e-8	0.500	0.1978	0.025	0.0093	0.758	0.2201	0.037	0.1152	0.050
Archive, k-nearest set	1.2e-9	**1.000**	**1.7e-4**	**0.960**	**3.9e-5**	**0.998**	0.0069	0.698	0.0035	**0.805**
Archive, global top-k	1.2e-9	**1.000**	**1.7e-4**	**0.960**	**3.9e-5**	**0.998**	0.0069	0.698	0.0035	**0.805**
MDN	1.2e-4	0.922	9.1e-4	0.148	0.0073	0.186	0.1224	0.071	0.0632	0.077
MCL-8	1.6e-7	**1.000**	2.9e-4	0.441	0.0476	0.131	0.0970	0.043	0.0649	0.059

and no fitted loss is evidence that Lemma 1 describes averaging itself rather than a property of neural regression. Second, retrieval is strong wherever the stored grid is dense enough, but it is not uniformly the best source: on ZDT3 it attains the lowest gap we measured, whereas on Adult-MLP the objective-trained point map's training-split gap of -0.0125 lies below global retrieval's 0.0035.

What this implies. The value of a learned source is therefore not accuracy on a densely tabulated preference grid but behaviour off that grid, together with a footprint that does not grow with the number of stored solutions: the archive has no trainable parameters, but it stores every retained solution and re-scalarizes and sorts the whole store at each query (Tables 11 and 12).

F.3 Symmetry-aware point estimation

Question. On the two weight-space tasks a decision is a neural network, and permuting hidden units while jointly flipping a unit's incoming weights, bias, and outgoing weights leaves the represented function unchanged. Teacher restarts can therefore land in numerically distant but functionally identical weight vectors, which raises a fair objection to both of our metrics: perhaps the squared-loss collapse is an artifact of averaging across a symmetry orbit, and perhaps the recall advantage merely counts the same function several times.

Setup. We canonicalize by matching hidden-unit signatures to a fixed reference orbit representative with the Hungarian algorithm, choosing each unit's sign from the sign of its inner product with the matched reference unit. The construction is verified numerically: a random signed permutation leaves both objectives unchanged to machine precision and canonicalization recovers the reference exactly. Endpoint regression on canonicalized targets is then ordinary L_2 regression in the quotient space, and every recall number on these tasks is recomputed with quotient distance for both the teacher classes and the candidate assignment.

Result. Table 9 reports both. Alignment removes roughly two fifths of the mean-collapse gap on each task and ensembling the aligned estimator removes a little more, but neither closes the point-policy recall gap. The quotient recomputation merges no teacher representatives at all, while changing the nearest assignment for about a fifth of digit-pair candidates.

What this implies. The objection is partly correct and does not overturn the conclusion: the known signed-permutation group accounts for a measurable share of the squared-loss gap, so we report quotient recall throughout rather than raw recall. It accounts for none of the recall ceiling, because that ceiling follows from output cardinality rather than from representation. Surviving classes may still contain unmodeled functional equivalences, so the objective-space diversity column is reported as a complementary check.

Table 9: Symmetry-aware point estimation on the two weight-space tasks. "Aligned" regresses on teacher endpoints canonicalized to a common signed-permutation orbit representative. Recall uses signed-permutation-quotient nearest assignment and the quality gate $\varepsilon_{\text{rec}} = 10^{-3}$; "Obj. div." is mean pairwise distance between served candidates in normalized objective space. Bold marks the best gap and recall per column.

	Digit-pair MTL (d=2404)			Adult-fair MLP (d=705)		
Policy	Ref. gap ↓	Q-recall ↑	Obj. div.	Ref. gap ↓	Q-recall ↑	Obj. div.
Endpoint (L_2)	0.5705	0	0	0.2190	9.8e-4	0
Endpoint, aligned	0.3516	0	0	0.1341	0	0
Endpoint ens.-8	0.4879	0	0.126	0.1820	0.006	0.082
Endpoint, aligned, ens.-8	0.3049	0	0.073	0.1141	7.8e-4	0.052
PHN-obj	**-0.0447**	**0.756**	0	**-0.0125**	0.797	0
MDN-16	0.0266	0.194	0.864	0.0242	0.186	0.752
Archive, global top-k	0.0069	0.698	0.262	0.0035	**0.805**	0.317

Table 10: Train-split versus held-out-data frozen-reference gap. "Train" is the number on the split the selector optimized; "Held-out" re-scores the same served decisions and the same frozen reference on disjoint data, averaged over five draws. Bold marks the lowest gap per column.

	Adult-fair (lin.)		Adult-fair (MLP)		Digit-pair	
Policy	Train	Held-out	Train	Held-out	Train	Held-out
Endpoint (L_2)	0.0024	0.0042	0.2190	0.1227	0.5705	0.5221
PHN-obj	5.9e-4	0.0028	-0.0125	0.0528	-0.0447	0.0056
Endpoint ens.-8	0.0011	0.0029	0.1820	0.0894	0.4879	0.4416
PHN-obj ens.-8	**1.7e-4**	**0.0025**	**-0.0165**	0.0552	**-0.0453**	**0.0042**
MCL-8	0.0011	0.0027	0.0649	0.0511	0.0970	0.1040
MDN	0.0037	0.0054	0.0632	0.0415	0.1224	0.1206
MDN-16	0.0026	0.0042	0.0242	**0.0275**	0.0266	0.0341
cVAE	0.0012	0.0027	0.1261	0.0310	0.3357	0.2998
Random search	0.1964	0.1891	0.2988	0.1912	1.2335	1.1656

F.4 HELD-OUT-DATA EVALUATION

Question. On the two data-derived tasks the training split defines the optimization problem: the teacher, the amortizer, and the serving-time selector all see only it. A reader is entitled to ask whether the reported separation is a property of the solution structure or merely of that split, and in particular whether PHN-obj's *negative* frozen-reference gap is real or an artifact of optimizing the very split the reference was built on.

Setup. We re-score the *stored* served decisions and the frozen reference under an evaluator built on disjoint data, so nothing is retrained and the comparison stays like for like. For digit-pair the held-out evaluator draws its pairs only from the 561 scikit-learn images that appear in no training pair, making the split disjoint at the image level rather than at the pair level; for Adult it draws disjoint rows from the 43,174 outside the 2048-row training subsample, encoded with the training statistics so the decision layout is unchanged. Results average five independent held-out draws, with the spread across draws reported separately from the spread across model seeds.

Result. Table 10 shows that the qualitative verdicts survive but one number does not. Every family still ties near zero on the linear Adult task, and the squared-loss collapse persists on both weight-space tasks, though at a smaller ratio than on the training split. Symmetry alignment helps *more* off-split than on it, which is consistent with alignment removing parameterization noise that would not have transferred anyway. The exception is PHN-obj: its negative gap turns positive off-split on both tasks, and on the Adult MLP it now trails the high-capacity generator.

What this implies. The negative gaps reported for objective-trained point maps in Table 2 should be read as train-split optimism against an approximate reference, not as beating the optimum, which is why the main text never rests any claim on their sign. The mechanism-level conclusions are unaffected, because they concern the ordering between supervision types and between cardinalities, and that ordering is what the held-out re-scoring preserves.

Table 11: Serving latency at k=64, in milliseconds. Entries are medians over 25 timed batch-one repeats after 10 excluded warm-up repeats; the end-to-end (E2E) column also gives the interquartile range in brackets. "Propose" is candidate construction and "Objective" is the common 64-call scoring pass; E2E additionally includes selection and is measured directly. A point row evaluates one model output but duplicates and scores it in all 64 slots; ensembles use eight model evaluations, multi-head maps one, and generators a batched conditional call. The archive row uses a synthetic A=128 pool. Measurements used an Intel Core i7-13700 CPU with one Torch/BLAS thread.

		ZDT3			Adult-fair (MLP)			Digit-pair		
Policy	k	Propose med.	Objective med.	E2E med. [IQR]	Propose med.	Objective med.	E2E med. [IQR]	Propose med.	Objective med.	E2E med. [IQR]
Endpoint / PHN-obj (single)	64	0.076	0.060	0.171 [0.007]	0.137	5.477	5.673 [0.410]	0.244	20.175	20.503 [0.435]
point map + noise	64	0.099	0.063	0.202 [0.029]	0.297	5.372	5.732 [0.274]	0.728	20.219	21.044 [0.349]
ensemble-8	64	0.397	0.070	0.517 [0.030]	0.658	5.435	6.148 [0.448]	1.175	20.780	22.006 [0.971]
multi-head-8	64	0.081	0.058	0.172 [0.007]	0.294	5.467	5.899 [0.379]	0.698	20.164	20.954 [0.396]
MDN (C=8)	64	0.126	0.061	0.226 [0.015]	0.654	5.505	6.238 [0.264]	1.865	20.238	22.168 [0.380]
MDN-16	64	0.143	0.062	0.241 [0.002]	1.560	5.406	7.068 [0.198]	4.616	20.141	24.886 [0.294]
cVAE	64	0.119	0.062	0.217 [0.020]	0.289	5.222	5.620 [0.229]	0.663	20.252	21.044 [0.468]
archive (global; synthetic A=128)	64	0.068	0.061	0.160 [0.007]	0.187	5.564	5.841 [0.360]	0.272	20.259	20.674 [0.480]
random search	64	0.029	0.056	0.114 [0.001]	0.152	5.114	5.323 [0.148]	0.449	20.328	20.853 [0.636]

F.5 Per-query serving cost

Question. The main text quotes end-to-end latencies but not what they are made of, and a serving contract is only useful if its overhead is understood in the right units. Three resources matter and the wall clock shows only the first: query latency, stored size, and the query volume at which amortizing pays for itself at all.

Setup and result. Table 11 decomposes batch-one latency into candidate construction, the common 64-call scoring pass, and selection. The decomposition is what carries the argument: scoring dominates, and it is charged identically to every policy by construction, so the interface change costs only the difference in candidate construction. A point policy's time includes 64 duplicate objective evaluations of its single decision, so it is not native one-point deployment latency. Table 12 reports trainable parameter counts; the archive has none but stores every retained solution. Table 13 solves $C_{\text{offline}} + Q\,C_{\text{serve}} = Q\,C_{\text{solve}}$ for the break-even query count Q, taking C_{solve} to be one projected scalarized solve timed on the same machine.

What this implies. Amortization pays off in the low hundreds of queries on all three tasks, which is the regime the contract targets, and the relative overhead of a candidate set shrinks as the objective gets more expensive because the common scoring pass grows for everyone. The break-even figures are an accounting illustration under the stated components rather than a benchmark, because the one-start online solve and the learned generator differ in quality and in what they return.

Table 12: Trainable parameter counts. "Networks" is the number of independently trained models behind one policy. The archive and random search have no trainable parameters; the archive instead stores every retained teacher solution together with its objective vector.

Policy	Networks	ZDT3	Adult-fair (MLP)	Digit-pair
Endpoint / PHN-obj (single)	1	0.04 M	0.12 M	0.34 M
point map + noise	1	0.04 M	0.12 M	0.34 M
ensemble-8	8	0.30 M	0.99 M	2.75 M
multi-head-8	1	0.06 M	0.76 M	2.51 M
MDN (C=8)	1	0.10 M	1.49 M	5.00 M
MDN-16	1	0.38 M	5.93 M	19.91 M
cVAE	1	0.04 M	0.22 M	0.65 M
archive (global; synthetic A=128)	0	0	0	0
random search	0	0	0	0

F.6 Sensitivity to an imperfect selector

Question. The whole contract rests on exact query-time scoring: a candidate set is only as good as the selector that ranks it. Where the objective is a simulator, a subsampled dataset, or a surrogate, the selector ranks by a noisy estimate, and the natural worry is that this erases the advantage a set was supposed to buy.

Table 13: Illustrative cost-only crossover against one $K{=}100$, one-restart online scalarized solve. Offline time is the recorded shared-teacher-plus-MDN total and serve time is the controlled batch-one query at $k{=}64$; quality and output interfaces differ from the online solve.

Task	Offline (s)	Serve, $k{=}64$ (ms)	Per-query solve (ms)	Break-even queries
ZDT3	7.6	0.23	44	172
Adult-fair (MLP)	29.6	6.24	82	389
Digit-pair	58.8	22.17	129	549

Setup. We measure the sensitivity offline, without retraining. For every stored candidate set we corrupt the *observed* normalized objectives with i.i.d. Gaussian noise of scale σ_{sel}, re-run the $\arg\min$, and then score the chosen candidate under the *true* objectives; $\sigma_{\mathrm{sel}} = 0.01$ is one percent in normalized objective units. Twenty noise draws are nested within each of the first five model seeds, so the seed remains the experimental unit. This quantity is distinct from the construction-noise multiplier of the point-plus-noise sequence, which is 0.5 for the sequence used in this study.

Result. Table 14 reports the outcome, and single-candidate policies are flat in σ_{sel} by construction. At one percent noise the candidate sets keep a large margin over the collapsed squared-loss map on every task, and their standing against the *strongest* point policy is unchanged from the noiseless protocol. At ten percent noise on ZDT3 the advantage disappears: the generator and the point map become statistically indistinguishable, with a paired confidence interval that straddles zero.

What this implies. The selector must resolve score differences on the scale of the gap the candidate set is meant to exploit, and ZDT3 is where that gap is smallest in absolute terms, which is exactly where noise erases it first. Selector accuracy is therefore a precondition of the contract rather than a free parameter, and it is stated as such in Section 6.

Table 14: Frozen-reference gap when the selector ranks candidates using noisy objectives. σ_{sel} is selector-objective noise in normalized units; it is distinct from the point-plus-noise sequence's construction multiplier, which is 0.5 here (the primary comparator uses multiplier 1). Values average twenty nested selector-noise draws within each of the first five model seeds; the seed, not the draw, is the experimental unit. Single-candidate policies are flat in σ_{sel}. Bold marks the lowest gap per column within each task.

		selector noise σ (normalized objective units)			
Task	Policy	exact	0.010	0.030	0.100
OmniTest	Endpoint (L_2)	0.0760	0.0760	0.0760	0.0760
	PHN-obj	3.8e-4	**3.8e-4**	**3.8e-4**	**3.8e-4**
	MCL-8	**2.2e-4**	6.4e-4	8.3e-4	0.0011
	MDN	5.2e-4	0.0016	0.0038	0.0149
	MDN-16	0.0040	0.0050	0.0080	0.0231
	Endpoint + noise	0.0103	0.0109	0.0141	0.0343
	Random search	0.0135	0.0140	0.0175	0.0365
ZDT3	Endpoint (L_2)	0.0502	0.0502	0.0502	0.0502
	PHN-obj	0.0377	0.0377	0.0377	0.0377
	MCL-8	0.0511	0.0512	0.0514	0.0526
	MDN	0.0067	0.0076	**0.0117**	0.0385
	MDN-16	**0.0064**	**0.0075**	**0.0117**	**0.0334**
	Endpoint + noise	0.0213	0.0227	0.0273	0.0391
	Random search	0.2786	0.2791	0.2830	0.3124
Digit-pair	Endpoint (L_2)	0.5492	0.5492	0.5492	0.5492
	PHN-obj	**-0.0454**	**-0.0454**	**-0.0454**	**-0.0454**
	MCL-8	0.0963	0.0964	0.0968	0.0999
	MDN	0.1234	0.1243	0.1265	0.1354
	MDN-16	0.0225	0.0240	0.0267	0.0368
	Endpoint + noise	0.4788	0.4804	0.4899	0.5226
	Random search	1.2344	1.2348	1.2379	1.2679

F.7 Retrospective constrained re-scoring of RE24

Question. Nothing in the paper so far tests the contract where a query can fail outright, because feasibility has been handled by the repair operator. RE24 offers a natural setting: its second objective

is the aggregated violation of four design constraints, so the benchmark as posed trades weight against violation and its reference front deliberately contains violating designs. The question is what a candidate set buys when the service must return a design that actually satisfies the constraints.

Setup. This is an offline selector-sensitivity study: every policy keeps its original bi-objective training, and we only re-normalize and re-select the already stored outputs. One geometric fact drives the setup. On a deterministic 801×801 grid, 91.1% of the box satisfies all four constraints, yet the lightest satisfying design is heavier than the suite's upper scaling anchor, so the entire satisfying region lies *outside* the reference box. Under the suite's own normalization, therefore, no method can return a satisfying design inside the reference box; this is a property of the benchmark, not of any solver. We therefore re-scale by fixed feasible-region anchors estimated from that grid and apply a feasibility-first selector that ranks only satisfying stored candidates.

Result. Table 15 reports feasible service rate as the primary quantity, because under this reading the residual objective is nearly single-objective: the violation term is zero for every admissible candidate. Single-candidate policies almost never return a satisfying design, whereas several candidate sets supply one for a large fraction of preferences. The penalized frozen-reference gap, which charges the normalized span whenever no admissible candidate exists, is reported as secondary.

What this implies. Where feasibility is a hard requirement rather than a repaired coordinate, output cardinality changes whether the service can answer at all, not merely how well. This is the sharpest form of the coverage argument, and it is retrospective: the policies were never trained for the constrained problem, so it bounds what the interface can recover rather than what a constraint-aware method would achieve.

Table 15: Feasibility-first re-scoring of stored RE24 candidates over five seeds. All serving policies retain their original bi-objective training. "Feasible candidates" is the fraction of the 64 stored slots satisfying all four design constraints; "feasible service" is the fraction of preferences for which the selector can return one. Bold marks the best feasible service and penalized gap.

Policy	Feasible candidates	Feasible service ↑	Penalized frozen-ref. gap ↓
Endpoint (L_2), single candidate	0.022	0.022 ± 0.049	0.978
PHN-obj, single candidate	0.000	0.000 ± 0.000	1.000
PCGRF, candidate set	0.155	0.169 ± 0.093	0.831
cVAE, candidate set	0.069	0.503 ± 0.043	0.497
MDN, candidate set	0.062	$\mathbf{0.838 \pm 0.150}$	**0.215**