LEARNING WHERE TO SEARCH: EVOLUTIONARY EMBEDDING SEARCH FOR HIGH-DIMENSIONAL BAYESIAN OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Bayesian optimization (BO) uses probabilistic models to optimize expensive black-box functions, guiding evaluations with predictions and their uncertainty. In high-dimensional BO (HDBO), fixed random embeddings reduce the number of search variables but leave the choice of bounded search region unoptimized for the objective. This region may therefore exclude good solutions that further BO evaluations cannot reach. To address this limitation, we propose Evolutionary Embedding Search (EES), which directly optimizes the embedding using objective evaluations. EES maps the same low-dimensional sample points through candidate embeddings and uses their objective values to guide evolution before running BO in the selected region. Our analysis motivates this matrix search: even within the same subspace, rotating a bounded embedding can change which solutions BO can reach. On synthetic benchmarks, EES achieves lower mean final log regret than random embedding search and evolution with shuffled scores under matched budgets. In real-world application benchmarks for rover trajectory planning and support-vector regression, EES also finds better solutions early in the subsequent BO run than direct baselines. The results show that optimizing the embedding, not only the points within it, can improve subsequent BO, and position EES as a direct, derivative-free mechanism for learning where HDBO should search.

1 INTRODUCTION

Bayesian optimization (BO) optimizes expensive black-box functions by fitting a probabilistic model to previous evaluations and using an acquisition function to choose the next input (Snoek et al., 2012; Frazier, 2018). It is useful when each evaluation requires a trial or simulation and reliable gradients are unavailable. In high-dimensional BO (HDBO), learning a useful model and optimizing the acquisition function both become more demanding. This has motivated methods that exploit structure in the objective, alongside improvements to conventional BO models (Binois & Wycoff, 2022; Hvarfner et al., 2024; Xu et al., 2025).

One approach is to search through a low-dimensional embedding. The optimizer chooses a small vector of coordinates and maps it to a full input, making low-dimensional search possible when the objective depends on relatively few directions (Wang et al., 2016). REMBO established the potential of random linear embeddings, and HeSBO and ALEBO improved their construction and geometry (Nayebi et al., 2019; Letham et al., 2020). The embedding also determines which inputs BO can visit. Choosing it well therefore matters twice: it reduces the optimization dimension and makes useful solutions accessible within the chosen bounds.

Existing work shows why this choice deserves attention. Bounded random embeddings can exclude an optimum, while clipping can distort the function seen by the surrogate (Wang et al., 2016; Letham et al., 2020). Improving the surrogate alone cannot make an excluded solution reachable. ALEBO addresses domain and kernel geometry, BAxUS expands nested embeddings, and EGORSE learns supervised directions from accumulated evaluations (Letham et al., 2020; Papenmeier et al., 2022; Priem et al., 2025). These approaches establish the importance of adapting the representation. We

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.

pursue a complementary decision: at a specified latent size, directly compare and optimize the bounded embedding matrices using the objective values they expose.

We propose Evolutionary Embedding Search (EES), a direct, derivative-free search over the embedding matrix. Within each generation, it maps the same low-dimensional sample points through a population of matrices and uses their objective values to select, combine, and perturb candidates. The selected matrix then defines the region for subsequent BO. This makes the region itself responsive to the objective and allows promising embeddings to guide further matrix proposals.

The resulting perspective treats a bounded embedding as an optimization variable. Even matrices spanning the same subspace can expose different solutions when their latent coordinates are bounded. Direct matrix search addresses this dependence by evaluating candidate regions before committing subsequent BO to one of them. Our contributions are:

- We characterize how the basis and latent bounds determine the region accessible to BO, providing a geometric foundation for searching the complete embedding matrix (Section 3.1).
- We introduce EES as an objective-guided method for selecting bounded embeddings. Its evolutionary search adapts the search region through direct evaluations, offering a derivative-free complement to random and supervised embedding construction (Section 3).
- We establish a benefit of objective-guided embedding selection for subsequent BO on the synthetic suite and demonstrate its potential on rover trajectory planning and support-vector regression. The application study also shows how latent size shapes optimization progress (Sections 4.2–4.4).

Sections 2–3 cover prior work and the method. Sections 4–5 present the evidence and research directions. Proofs and extended results appear in the appendix.

2 Related work

High-dimensional BO methods differ in what they simplify: the variables being searched, the structure of the objective, or the model used to predict it. This distinction clarifies where embedding selection enters the problem and which alternatives need no separate embedding search.

Reducing the search space. REMBO uses random linear embeddings under a low effective-dimension assumption (Wang et al., 2016). HeSBO uses sparse hashing to construct the map (Nayebi et al., 2019), while ALEBO studies how the embedded domain and kernel affect optimization (Letham et al., 2020). These results show that reducing the number of coordinates is only part of the design: the mapped domain must also contain useful solutions. BAxUS expands nested embeddings, and DSEBO adapts random embeddings when the effective dimension is unknown (Papenmeier et al., 2022; Qian et al., 2026). X-REGO solves a sequence of reduced problems under multiple random embeddings (Cartis et al., 2023).

Learning which structure matters. Some methods estimate useful variables or directions from observations. SAASBO places a sparse prior on coordinate relevance (Eriksson & Jankowiak, 2021). Projection-pursuit models learn combinations of input variables (Li et al., 2016), and GP-prior variational autoencoders learn a nonlinear latent representation (Ramchandran et al., 2025). EGORSE alternates supervised and random linear embeddings, relearning supervised directions from accumulated objective observations (Priem et al., 2025). GIT-BO forms subspaces from a foundation model's predictive gradients (Yu et al., 2026). Additive BO instead represents the objective as a sum of simpler functions, easing modeling and acquisition search when this decomposition is appropriate (Kandasamy et al., 2015; Rolland et al., 2018).

Optimizing without a fixed global embedding. TuRBO-1 adapts local search regions (Eriksson et al., 2019). Line-search and covariance-adaptation methods change the directions explored during optimization (Kirschner et al., 2019; Ngo et al., 2024). BOIDS combines incumbent-guided lines with expanding random embeddings based on the BAxUS rule (Ngo et al., 2025). Conventional GP models can also remain effective in high dimensions when their priors, initialization, or kernels are

chosen appropriately (Hvarfner et al., 2024; Xu et al., 2025; Papenmeier et al., 2025; Doumont et al., 2026). These methods adapt the optimizer or its model while using evaluations directly to improve the objective.

Evolutionary search over representations. Evolutionary algorithms offer derivative-free proposals for structured search variables (Eiben & Smith, 2015). A related preprint applies differential evolution and QR repair to basis-invariant subspace objectives (Lesniewski, 2025). For such objectives, different orthonormal bases of the same subspace are equivalent. A bounded latent box introduces a different selection problem because its image can depend on the basis.

These approaches address different parts of representation design. Random embeddings offer inexpensive construction, but their usefulness depends on the sampled map and domain bounds. Adaptive dimension expands coverage, while supervised embedding methods infer directions through a fitted representation. Local methods and improved GP models adapt how points are selected without separately optimizing a bounded embedding matrix. EES addresses the specific choice left by a prescribed latent box: which matrix makes that region useful for the objective? It scores candidate matrices directly on a shared sample design and uses those scores to guide evolutionary search. This optimizes both the subspace and the orientation of the bounded region, providing an objective-guided alternative to drawing a matrix once and then optimizing only its coordinates.

3 Evolutionary Embedding Search

EES first searches for an embedding in which low objective values are easier to find, then runs Bayesian optimization in that embedding. The embedding-search stage uses an evolutionary algorithm: it maintains a population of candidate matrices, scores their sampled objective values, and forms new candidates from those that score well (Eiben & Smith, 2015). We describe the search variable before introducing this selection procedure.

3.1 The embedding as a search variable

We minimize a black-box function f over $\mathcal{X} = [-1, 1]^D$. Any finite rectangular domain can be put in this form by rescaling its coordinates. An embedding represents a point by d latent coordinates $z \in \mathcal{Z}_s = [-s, s]^d$, where $1 \le d \le D$ and $s > 0$. Its matrix $B \in \mathbb{R}^{D \times d}$ maps these coordinates into the original space:

$$h_B(z) = f(\phi_B(z)), \qquad \phi_B(z) = \mathrm{clip}(Bz, -\mathbf{1}, \mathbf{1}). \tag{1}$$

Clipping replaces an out-of-bounds coordinate by its nearest bound. We require $B^\top B = I_d$, so its columns are orthonormal and $\|Bz\|_2 = \|z\|_2$ before clipping. This fixes the matrix scale, keeps the latent directions independent, and lets s control the extent of the search.

Figure 1 illustrates the choice of embedding using squared distance to a target at $(0.6, 0.6)$. If BO can move only along the horizontal line, it cannot reach the target. Tilting the line opens a route to it. The same issue can arise without changing the subspace: both squares in the middle panel lie in the same plane, yet their different orientations determine whether the target lies inside the search region. The right panel then shows how objective values at mapped sample points can distinguish the two line embeddings. EES uses this kind of objective feedback to choose among candidate regions.

More generally, an orthogonal change of basis preserves $B\mathcal{Z}_s$ exactly only when it permutes coordinates or changes their signs. Clipping can make additional images coincide. Proposition 1 and its proof appear in Appendix B.

EES searches the full matrix, allowing both the span and its basis to change. Orthonormality leaves $Dd - d(d+1)/2$ continuous degrees of freedom (Absil et al., 2008). This representation lets the search adapt the orientation of the bounded region as well as the directions it spans.

3.2 Learning an embedding from objective evaluations

A candidate is promising if sampled points in its image have low objective values. Each generation uses one scrambled Sobol design of m latent points to spread evaluations across the latent box (Matoušek, 1998; Owen, 1998). All N candidates use these same coordinates, so the comparison

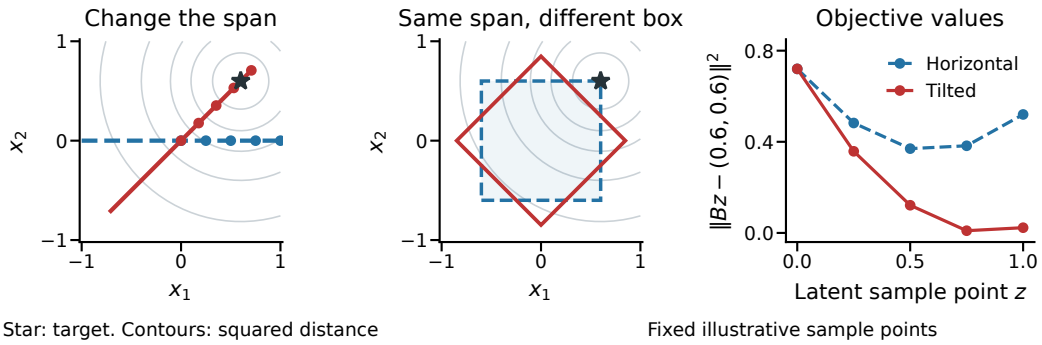

Figure 1: Choosing a bounded embedding for the objective $f(x) = \|x - (0.6, 0.6)\|_2^2$. The target has value zero, so an embedding that reaches it permits the best possible solution. Left: tilting a line with $s = 1$ makes the target reachable. Middle: rotating a square with $s = 0.6$ can exclude the target even though both bases span the same plane. Right: mapping five illustrative points in $[0, 1]$ through the two lines gives mean values 0.495 and 0.246. These one-sided probes explain the scoring idea.

Algorithm 1 Evolutionary Embedding Search

Require: Objective f, $d, s, N, G, m, k, \lambda$, evolution settings $n_e, \tau, p_c, p_m, \sigma$, BO budget T, initialization size n_0

1: Initialize N Gaussian matrices by reduced QR and set archive $\mathcal{A} \leftarrow \varnothing$.
2: **for** $g = 1, \ldots, G$ **do**
3: Draw one m-point scrambled Sobol design in $\mathcal{Z}_s$.
4: Evaluate every embedding on this design and store each matrix and its raw (T_k, C) in $\mathcal{A}$.
5: Compute the population scores using Eq. (2).
6: **if** $g < G$ **then**
7: Retain the n_e best matrices and select parent pairs by size-τ tournaments.
8: Fill the population by columnwise crossover (p_c), QR, columnwise mutation (p_m, σ), and QR.
9: **end if**
10: **end for**
11: Rescale the raw components over $\mathcal{A}$ and choose B^* with the lowest archive score.
12: Run subsequent BO for T evaluations of h_{B^*}, initialized with n_0 new observations. Return its best point.

varies the embedding while holding the sample design fixed. For each matrix, we evaluate the mapped points and compute $T_k(B)$, the mean of the k smallest responses. Averaging several good values makes the score depend on more than one sampled minimum. A clipping penalty $C(B)$ adds the fraction of clipped coordinates to the mean absolute clipping displacement over all sampled coordinates. The score is

$$q(B) = (1 - \lambda)\widetilde{T}_k(B) + \lambda\widetilde{C}(B), \qquad 0 \le \lambda \le 1, \tag{2}$$

where each component is rescaled to $[0, 1]$ using its range among the current candidates. Lower scores are preferred. All sample points are fixed before evaluating the candidates.

Evolution retains the best matrices, selects parents by score, and combines their columns before adding random perturbations. Reduced QR restores orthonormality after crossover and again after mutation. Algorithm 1 summarizes these derivative-free proposals. Appendix A defines their parameters and exact order.

The final matrix is selected from all evaluated entries, including earlier generations. We recompute Eq. (2) from their raw components using one set of archive-wide ranges. Each evaluation of a retained matrix remains a separate entry. Figure 5 summarizes the information flow, and Appendix A gives the normalization details.

3.3 Bayesian optimization in the selected embedding

Once B^* is fixed, subsequent BO models h_{B^*} with a Gaussian process and chooses latent points by logarithmic expected improvement (Ament et al., 2023). It starts from n_0 new uniformly sampled points and refits the model after each observation. Its observations and best-so-far value come entirely from this stage. This initialization tests the selected embedding under the same BO conditions used for the embedding controls. Section 5 explains the design choice, and Appendix G reports a separate study of observation reuse.

With G generations of N matrices, m evaluations per matrix, and T subsequent BO evaluations including initialization, the total evaluation budget is

$$B_{\text{paid}} = GNm + T. \tag{3}$$

The experimental settings and corresponding evaluation counts are given in Section 4.

4 Experiments

We first test whether objective feedback improves the embedding used by subsequent BO. We then compare the resulting optimization curves with direct methods and examine latent size in two applications. Each curve shows the best objective value observed so far. Matched controls assess embedding selection, while direct-method comparisons describe the progress obtained after that selection.

4.1 Experimental setup

The synthetic suite covers five familiar function families: local minima in Ackley, multiple basins in Branin and Hartmann, a curved valley in Rosenbrock, and a full-dimensional quadratic control in Sphere (Surjanovic & Bingham, 2013; Letham et al., 2020; Papenmeier et al., 2022). Two active-coordinate scales of Rosenbrock give six objectives. Shifts remove a shared central optimum, and rotations distribute the active directions across coordinates. Branin and Hartmann also use smooth domain warps. Appendix C.1 defines these transformed objectives and their roles. The control study uses ten transformed instances per objective and five seeds per instance. The method comparison uses three separate instances per objective and three seeds per instance.

Table 1 gives the shared settings and evaluation budgets. All methods minimize the same objective within each comparison. For synthetic functions we plot the logarithm of the difference between the best observed value and the known optimum. A small fixed offset makes it finite at zero (Appendix C.3). For applications we plot the original objective. Lower curves are better. Axes labeled "Optimization evaluations" count subsequent BO evaluations, including initialization. Total-cost plots also count embedding search. Captions specify the aggregation and uncertainty.

Table 1: EES settings. The symbol s sets the latent half-width in normalized coordinates. Each search generation evaluates N embeddings at m points each. The BO budget includes its initial evaluations.

Symbol	Meaning	Controls	Synthetic comparison	Applications
D	Original dimension	50	50	60 / 388
d	Latent dimension	5	5	5, 10, 20, 30
s	Latent half-width	1	1	1, 2, 3
G	Search generations	5, 10, 15, 20	20	20
N	Population size	10	10	10
m	Sample points per embedding	512	512	512
k	Best values in the score	5	5	5
λ	Clipping-penalty weight	0.05	0.05	0.05
n_0	Initial BO evaluations	5	5	5
T	Total BO calls, including n_0	100	100	100

EES and its embedding controls use equal embedding-search budgets and the same subsequent GP–LogEI optimizer, with five new initial points. Direct BO baselines use the subsequent-BO budget

without a preceding embedding search, so their total cost is smaller. Their initialization follows each method's implementation. Separate Sobol references spend the full EES budget directly in the original domain. Appendix C.2 gives the baseline implementations and initialization counts, and Table 4 details the cost at each search length.

The common subsequent-BO budget measures how quickly a selected embedding supports improvement. We use $T = 100$ for the principal studies. Appendix G separately reports an earlier observation-reuse study with its own protocol and horizon.

4.2 Does feedback improve embedding selection?

We compare EES with two controls. The *random pool* draws a new matrix for each embedding evaluation and applies the same final selection score. EES instead spends some evaluations on retained elites. The *shuffled-score* control retains the evolutionary operators but randomly reassigns scores before reproduction. Its final selection uses the correct measurements. With equal budgets, these controls test the benefit of feedback-guided selection as a complete policy.

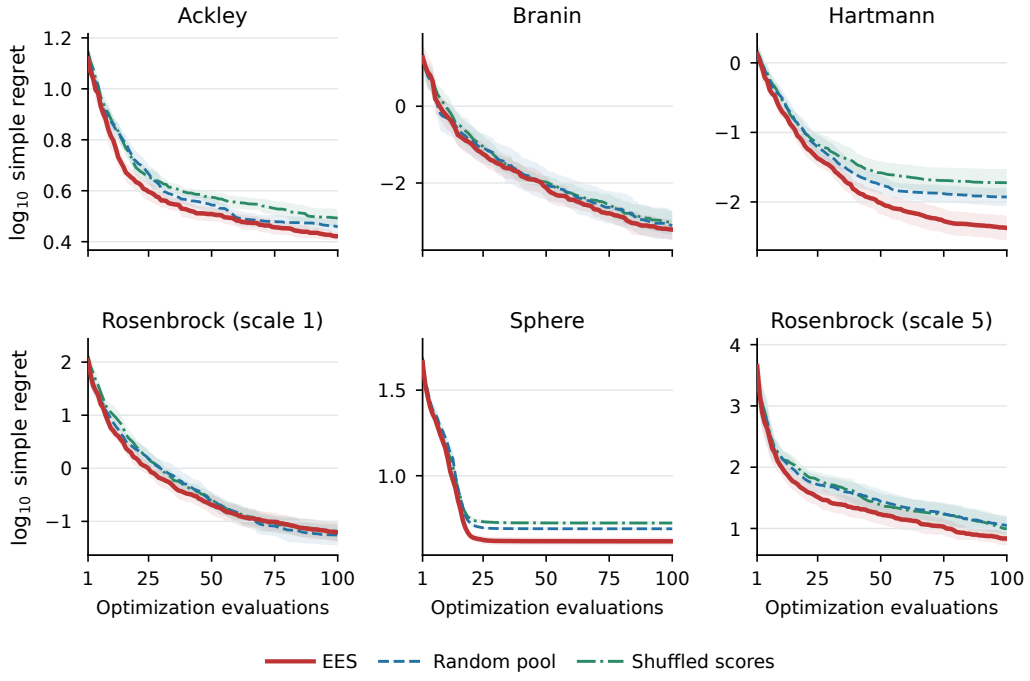

Figure 2: Subsequent BO after twenty search generations. All three methods use equal embedding-search and BO budgets (Table 1). Lines average ten transformed-instance means, each based on five seeds. At each evaluation, shaded bands give a 95% confidence interval for the mean across instances. Initialization is included.

Figure 2 shows a clear separation on Hartmann, where EES continues to improve as the control curves level off. On the larger-scale Rosenbrock problem, all three methods improve and EES maintains a lower mean curve. On Branin, all three decrease rapidly and remain closer together. Sphere improves mainly near the start, while Ackley shows a smaller persistent advantage for EES. Table 2 confirms the suite-average benefit: both simultaneous upper bounds are below zero after accounting for the eight control-and-budget contrasts. The smaller-scale Rosenbrock and Branin differences are weaker or change over the run. Appendix D reports all four search lengths and the fixed-suite uncertainty calculation.

Additional embedding search shows smaller mean gains after ten generations, but the paired intervals do not establish an optimal length (Figure 8 and Table 8).

Table 2: Primary selection result after twenty generations and subsequent BO. Differences are EES minus control in mean final $\log_{10}$ regret, averaged equally over six objectives. Negative values favor EES. Approximate simultaneous 95% upper bounds cover all eight tested control-and-budget contrasts (Appendix C.3).

Matched control	Mean difference	Simultaneous upper bound
Random pool	−0.136	−0.0170
Shuffled scores	−0.196	−0.0776

4.3 How does subsequent BO compare with direct optimization?

We next compare EES with adaptive embeddings BAxUS and DSEBO (Papenmeier et al., 2022; Qian et al., 2026), TuRBO-1 (Eriksson et al., 2019), Vanilla BO (Hvarfner et al., 2024), spherical linear BO (Doumont et al., 2026), MSR (Papenmeier et al., 2025), a standard Matérn GP (Xu et al., 2025), CMA-BO (Ngo et al., 2024), and Sobol sampling (Matoušek, 1998). These methods cover adaptive embeddings, local search, and direct modeling in the original space. Appendix C.2 specifies the implementations.

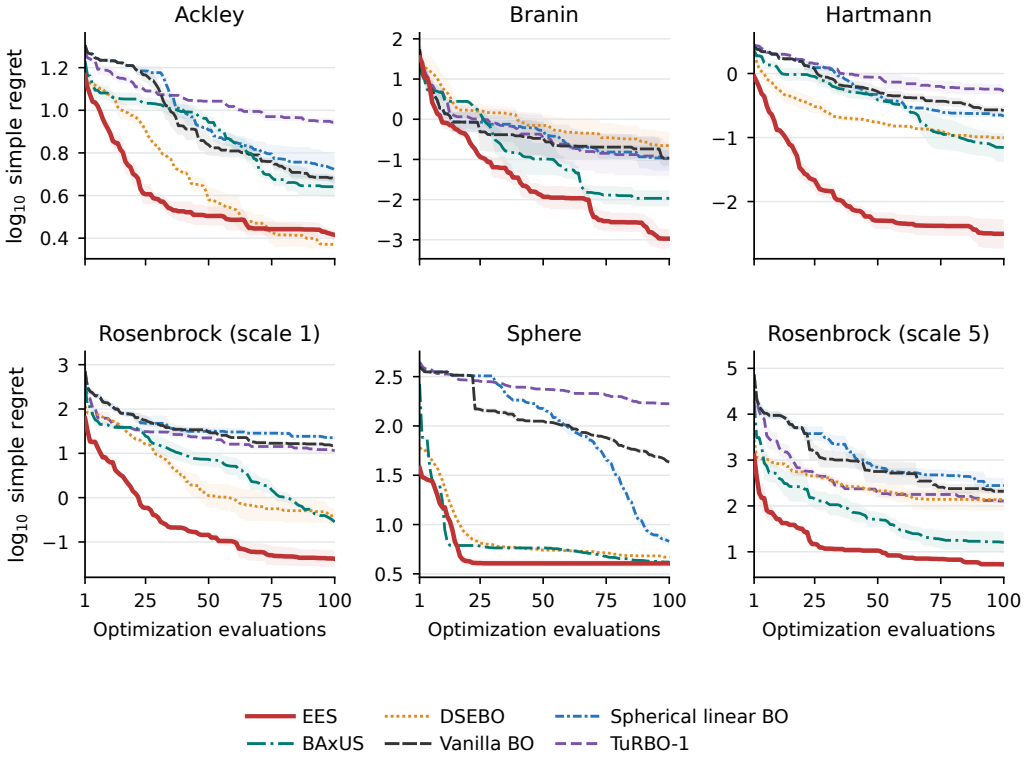

Figure 3: Synthetic comparison at $D = 50$. Figure 10 includes all ten methods. Lines average three transformed-instance means, each based on three seeds. Bands show one standard error across instances. EES is plotted during subsequent BO and additionally uses the embedding-search budget in Table 1. Direct methods use only the displayed budget.

EES has the lowest mean final log regret on five of the six objectives, while DSEBO finishes best on Ackley (Figure 3). Its Branin and Hartmann curves continue to descend after several direct methods level off, finishing more than one log-regret unit below the best direct mean. Both Rosenbrock variants also show lower EES curves by the end of the run. Sphere reaches a similar final level for EES and BAxUS (Table 9). These results describe the progress enabled by the selected embeddings under the larger total EES budget. Method-specific initialization also contributes to the early trajectories. Complete curves, descriptive ranks, and instance-level differences appear in Appendix E.

4.4 How does EES perform on the applications?

We use two established application benchmarks with different decision structures: spatial waypoint selection in rover trajectory planning and joint feature scaling and hyperparameter tuning in support-vector regression (Wang et al., 2018; Doumont et al., 2026). They test embedding search without supplying a known low-dimensional generative model. Rover-60 optimizes 30 planar waypoints in a simulated terrain-cost objective. SVR-388 tunes 385 input scaling factors and three regression hyperparameters on CT-slice localization data (Graf et al., 2011). We retain the fixed 250-example training and 250-example validation split used by the spherical-linear BO benchmark implementation, keeping the objective consistent across methods. Rover evaluations include small waypoint perturbations, and SVR reports the repeatedly queried validation RMSE. Appendix F.1 explains the noise, patient-identifier and normalization conventions.

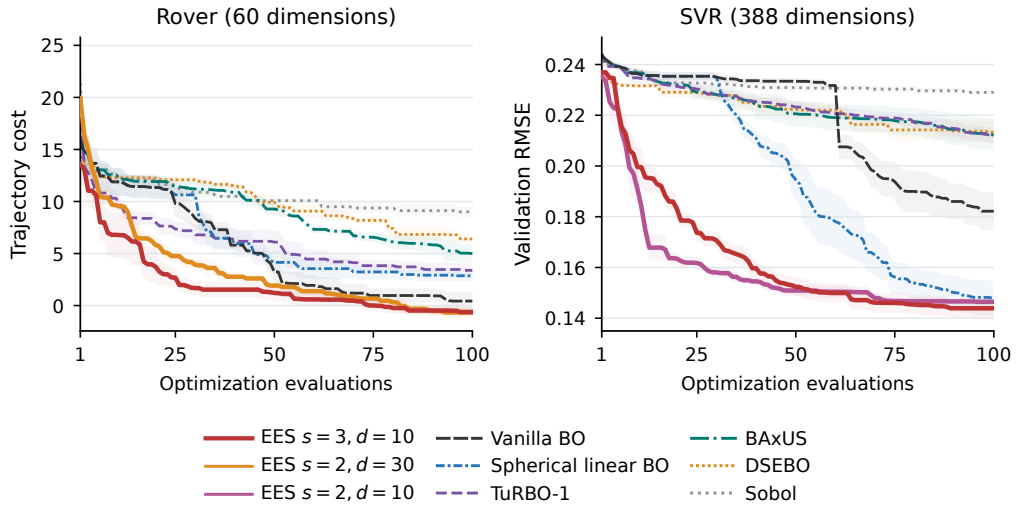

Figure 4: Application curves for EES settings selected from the size study. Lines average five runs, and bands show one standard error. EES is plotted during subsequent BO and additionally uses the embedding-search budget in Table 1. Direct methods use only the displayed budget.

At the selected $(s, d) = (3, 10)$ setting, EES makes rapid progress on both applications (Figure 4). Its mean best-so-far value is lower than all six direct methods throughout the displayed Rover run and from call five onward on SVR. On Rover, the EES curve falls sharply before improving more gradually. On SVR, its early decrease is followed by a flatter period while spherical linear BO continues to improve. Against the direct method with the lowest mean final value—Vanilla BO on Rover and spherical linear BO on SVR—EES has a lower curve mean on all five paired seeds. The curve mean averages the best-so-far value across subsequent BO. EES wins three of five final-value pairs on each task (Table 13). These are descriptive results at settings chosen from the same size study, with unequal total budgets and method-specific initialization.

The full grid shows why latent size matters (Figure 12, Appendix F). Moving from $(1, 5)$ to $(3, 10)$ changes the mean final Rover cost from 7.765 to -0.617, and SVR RMSE from 0.2067 to 0.1439. Rover's lowest observed final mean occurs at $(2, 30)$ and SVR's at $(3, 10)$. The largest tested setting, $(3, 30)$, gives higher final means than these respective settings. A larger latent box makes more points accessible for a fixed matrix, but also gives BO a larger region to explore with the same evaluation budget. The grid varies both the learned matrix and latent size, motivating adaptive choices that balance coverage and optimization difficulty (Appendix B.5).

Subsequent BO also improves on the best recorded embedding-search value in five of five Rover runs and four of five SVR runs at $(3, 10)$ (Table 15). The selected region therefore supports additional improvement beyond the evaluations used to select it.

5 DISCUSSION

Objective feedback selects embeddings that support better subsequent BO, most clearly on Hartmann and the larger-scale Rosenbrock problem. Branin and the smaller-scale Rosenbrock problem show weaker or changing differences. Hartmann improves with fewer latent than effective coordinates, suggesting that a useful search region can be valuable without recovering the full active subspace. Fixed-subspace comparisons could test the role of basis orientation. The restriction-error analysis in Appendix B.6 further separates region quality from the optimization still needed within it.

Starting subsequent BO with new observations tests the selected embedding independently of inheriting a favorable search observation. Reuse remains promising. An earlier study found mixed gains from selected-embedding observations and poor results from cross-embedding transfer (Appendix G). Its different search score and initialization policy motivate a dedicated study of coordinate-faithful reuse for EES. Independently reassessing the winning embedding could also address optimism from repeatedly evaluating retained matrices. The present controls support the complete selection policy, leaving individual operators and score components for targeted ablations.

Reducing the large embedding-search budget is the main practical priority. The gains in subsequent BO come with a higher total evaluation cost than the direct adaptive methods (Table 1). At $(s, d) = (1, 5)$, equal-cost Sobol also gives a lower mean Rover cost (Table 14). Cheaper preliminary evaluations, adaptive allocation among candidates, and transfer across related tasks could turn selection gains into end-to-end efficiency. Parallel evaluation can additionally reduce elapsed time when the objective permits it.

The application grid motivates task-dependent latent bounds and dimension, with independent assessment of its selected settings. In the reduced-budget follow-up, EES beats equal-cost Sobol in every run, but its final differences from the random pool remain unresolved and spherical linear BO wins all ten final SVR comparisons (Appendix F.6). Both budget and seed cohort change, calling for a matched-cohort budget study. Combining adaptive allocation with progressive dimension expansion, as in BAxUS (Papenmeier et al., 2022), could help choose (s, d) during optimization.

Higher-dimensional results also motivate dimension-scaled proposals and adaptive latent dimension. BAxUS wins most pairs at $D = 500$ and $D = 1000$, and TuRBO-1 wins 17 of 18 at $D = 1000$ (Appendix E.1). Several conditions change together, leaving their individual effects unresolved. More broadly, combining objective-guided embedding search with adaptive representations and improved high-dimensional GP models (Papenmeier et al., 2022; Hvarfner et al., 2024; Doumont et al., 2026) could strengthen both region selection and optimization within it. Equal-total-budget comparisons, longer subsequent BO runs, and independent reevaluation of noisy returned solutions would establish where that combination is most useful.

6 CONCLUSION

EES uses objective feedback to learn where HDBO should search. Its geometric motivation is simple: a bounded embedding determines which solutions BO can reach, so the embedding itself is an important optimization choice. Matched synthetic controls show that evolutionary selection improves subsequent BO on the tested suite, and the application study demonstrates the potential of selecting an appropriate search region and latent size. Making this benefit economical and reliable across tasks calls for adaptive search budgets, task-dependent latent dimensions, and selective observation reuse. These directions build on the central finding that learning the embedding can improve the optimization carried out within it.

AI USE STATEMENT

Generative-AI tools assisted experimental design and implementation, mathematical derivations and proof checking, analysis of recorded results, literature search, plotting, drafting, and editing. The authors remain responsible for verifying the analysis, claims, citations, and final text.

Reproducibility statement

The appendix details the algorithm and experimental protocols. The anonymous supplements provide experiment code, fixed configurations, software requirements, and instructions for running the studies. A separate results-and-replay package supplies archived numerical records that, together with the code, allow readers to check the reported results and regenerate tables and figures without new objective evaluations. The guides describe external baseline sources, the SVR dataset, setup checks, and the scope of the archived evidence and reconstructed historical reuse implementation.

References

P.-A. Absil, Robert Mahony, and Rodolphe Sepulchre. *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, 2008. doi: 10.1515/9781400830244.

Sebastian Ament, Samuel Daulton, David Eriksson, Maximilian Balandat, and Eytan Bakshy. Unexpected improvements to expected improvement for Bayesian optimization. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/419f72cbd568ad62183f8132a3605a2a-Abstract-Conference.html.

Mickael Binois and Nathan Wycoff. A survey on high-dimensional Gaussian process modeling with application to Bayesian optimization. *ACM Transactions on Evolutionary Learning and Optimization*, 2(2), 2022. doi: 10.1145/3545611. URL https://doi.org/10.1145/3545611.

Coralia Cartis, Estelle M. Massart, and Adilet Otemissov. Bound-constrained global optimization of functions with low effective dimensionality using multiple random embeddings. *Mathematical Programming*, 198(1):997–1058, 2023. doi: 10.1007/s10107-022-01812-9. URL https://doi.org/10.1007/s10107-022-01812-9.

Colin Doumont, Donney Fan, Natalie Maus, Jacob R. Gardner, Henry Moss, and Geoff Pleiss. We still don't understand high-dimensional Bayesian optimization. In *Proceedings of the 29th International Conference on Artificial Intelligence and Statistics*, volume 300 of *Proceedings of Machine Learning Research*, pp. 2629–2637, 2026. URL https://proceedings.mlr.press/v300/doumont26a.html.

Agoston E. Eiben and Jim Smith. From evolutionary computation to the evolution of things. *Nature*, 521:476–482, 2015. doi: 10.1038/nature14544. URL https://doi.org/10.1038/nature14544.

David Eriksson and Martin Jankowiak. High-dimensional Bayesian optimization with sparse axis-aligned subspaces. In *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence*, volume 161 of *Proceedings of Machine Learning Research*, pp. 493–503. PMLR, 2021. URL https://proceedings.mlr.press/v161/eriksson21a.html.

David Eriksson, Michael Pearce, Jacob Gardner, Ryan D. Turner, and Matthias Poloczek. Scalable global optimization via local Bayesian optimization. In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/6c990b7aca7bc7058f5e98ea909e924b-Abstract.html.

Peter I. Frazier. A tutorial on Bayesian optimization. In *INFORMS Tutorials in Operations Research*, pp. 255–278. INFORMS, 2018. doi: 10.1287/educ.2018.0188. URL https://doi.org/10.1287/educ.2018.0188.

F. Graf, H.-P. Kriegel, M. Schubert, S. Poelsterl, and A. Cavallaro. Relative location of CT slices on axial axis, 2011. URL https://doi.org/10.24432/C5CP6G. Dataset.

Carl Hvarfner, Erik Orm Hellsten, and Luigi Nardi. Vanilla Bayesian optimization performs great in high dimensions. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 20793–20817. PMLR, 2024. URL https://proceedings.mlr.press/v235/hvarfner24a.html.

Kirthevasan Kandasamy, Jeff Schneider, and Barnabas Poczos. High dimensional Bayesian optimisation and bandits via additive models. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 295–304. PMLR, 2015. URL https://proceedings.mlr.press/v37/kandasamy15.html.

Johannes Kirschner, Mojmir Mutny, Nicole Hiller, Rasmus Ischebeck, and Andreas Krause. Adaptive and safe Bayesian optimization in high dimensions via one-dimensional subspaces. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3429–3438. PMLR, 2019. URL https://proceedings.mlr.press/v97/kirschner19a.html.

Andrew Lesniewski. Differential evolution for Grassmann manifold optimization: A projection approach, 2025. URL https://arxiv.org/abs/2503.21984. Preprint.

Ben Letham, Roberto Calandra, Akshara Rai, and Eytan Bakshy. Re-examining linear embeddings for high-dimensional Bayesian optimization. In *Advances in Neural Information Processing Systems*, volume 33, pp. 1546–1558, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/10fb6cfa4c990d2bad5ddef4f70e8ba2-Abstract.html.

Chun-Liang Li, Kirthevasan Kandasamy, Barnabas Poczos, and Jeff Schneider. High dimensional Bayesian optimization via restricted projection pursuit models. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51 of *Proceedings of Machine Learning Research*, pp. 884–892. PMLR, 2016. URL https://proceedings.mlr.press/v51/li16e.html.

Jiří Matoušek. On the L_2-discrepancy for anchored boxes. *Journal of Complexity*, 14(4):527–556, 1998. doi: 10.1006/jcom.1998.0489.

Amin Nayebi, Alexander Munteanu, and Matthias Poloczek. A framework for Bayesian optimization in embedded subspaces. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 4752–4761. PMLR, 2019. URL https://proceedings.mlr.press/v97/nayebi19a.html.

Lam Ngo, Huong Ha, Jeffrey Chan, Vu Nguyen, and Hongyu Zhang. High-dimensional Bayesian optimization via covariance matrix adaptation strategy. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=eTgxr7gPuU.

Lam Ngo, Huong Ha, Jeffrey Chan, and Hongyu Zhang. BOIDS: High-dimensional Bayesian optimization via incumbent-guided direction lines and subspace embeddings. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(18):19659–19667, 2025. doi: 10.1609/aaai.v39i18.34165. URL https://ojs.aaai.org/index.php/AAAI/article/view/34165.

Art B. Owen. Scrambling Sobol' and Niederreiter–Xing points. *Journal of Complexity*, 14(4): 466–489, 1998. doi: 10.1006/jcom.1998.0487.

Leonard Papenmeier, Luigi Nardi, and Matthias Poloczek. Increasing the scope as you learn: Adaptive Bayesian optimization in nested subspaces. In *Advances in Neural Information Processing Systems*, volume 35, pp. 11586–11601, 2022. doi: 10.52202/068431-0842. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/4b7439a4ab0b8e4bcb4e2412c6a10a58-Abstract-Conference.html.

Leonard Papenmeier, Matthias Poloczek, and Luigi Nardi. Understanding high-dimensional Bayesian optimization. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 47902–47923. PMLR, 2025. URL https://proceedings.mlr.press/v267/papenmeier25a.html.

Rémy Priem, Youssef Diouane, Nathalie Bartoli, Sylvain Dubreuil, and Paul Saves. High-dimensional Bayesian optimization using both random and supervised embeddings. *AIAA Journal*, 63(1):162–173, 2025. doi: 10.2514/1.J063488. URL https://doi.org/10.2514/1.J063488.

Hong Qian, Xiang Shu, Xiang Xia, Xuhui Liu, Yangde Fu, Bei Liang, Huibin Wang, and Liang Dou. Automated random embedding for practical Bayesian optimization with unknown effective dimension, 2026. URL https://arxiv.org/abs/2605.23473. arXiv preprint arXiv:2605.23473.

Siddharth Ramchandran, Manuel Haussmann, and Harri Lähdesmäki. High-dimensional Bayesian optimisation with Gaussian process prior variational autoencoders. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/7b6e1905ebe2caa41870fcdcb9898243-Abstract-Conference.html.

Paul Rolland, Jonathan Scarlett, Ilija Bogunovic, and Volkan Cevher. High-dimensional Bayesian optimization via additive models with overlapping groups. In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pp. 298–307. PMLR, 2018. URL https://proceedings.mlr.press/v84/rolland18a.html.

Jasper Snoek, Hugo Larochelle, and Ryan P. Adams. Practical Bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems*, volume 25, 2012. URL https://proceedings.neurips.cc/paper/2012/hash/05311655a15b75fab86956663e1819cd-Abstract.html.

Sonja Surjanovic and Derek Bingham. Virtual library of simulation experiments: Test functions and datasets. Simon Fraser University, 2013. URL https://www.sfu.ca/~ssurjano/optimization.html. Accessed September 12, 2026.

Zi Wang, Clement Gehring, Pushmeet Kohli, and Stefanie Jegelka. Batched large-scale Bayesian optimization in high-dimensional spaces. In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pp. 745–754. PMLR, 2018. URL https://proceedings.mlr.press/v84/wang18c.html.

Ziyu Wang, Frank Hutter, Masrour Zoghi, David Matheson, and Nando de Freitas. Bayesian optimization in a billion dimensions via random embeddings. *Journal of Artificial Intelligence Research*, 55:361–387, 2016. doi: 10.1613/jair.4806. URL https://jair.org/index.php/jair/article/view/10983.

Zhitong Xu, Haitao Wang, Jeff M. Phillips, and Shandian Zhe. Standard Gaussian process is all you need for high-dimensional Bayesian optimization. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=kX8h23UG6v.

Rosen Ting-Ying Yu, Cyril Picard, and Faez Ahmed. GIT-BO: High-dimensional Bayesian optimization using tabular foundation models. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/b54bafb92da25f09ac367cff46678992-Abstract-Conference.html.

A EMBEDDING SEARCH AND OPTIMIZATION DETAILS

Algorithm 1 separates the evaluation of candidate embeddings from the BO run in the selected matrix. This section specifies the score and evolutionary operators. Numerical settings are given with the experiments, rather than being part of the algorithm definition.

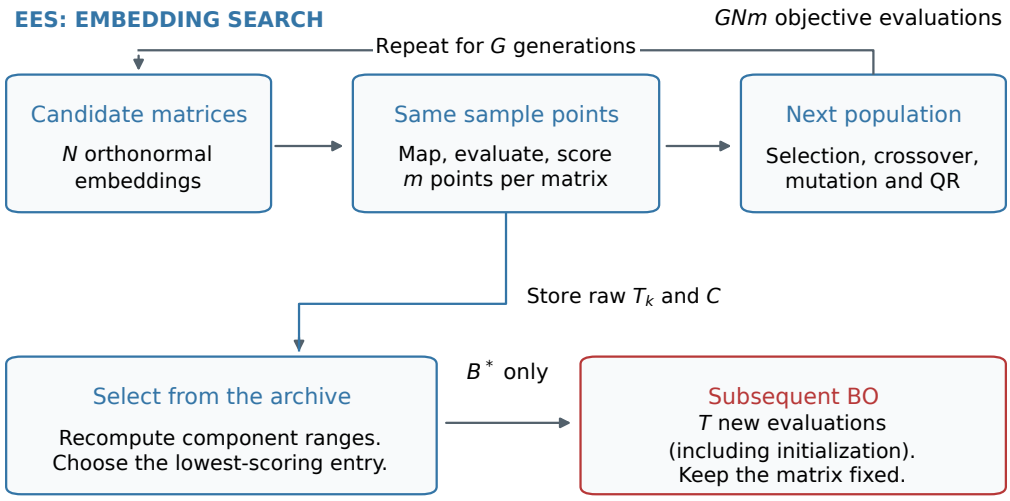

Figure 5: Information flow in EES. Each generation maps the same latent sample points through every candidate embedding. Scores guide reproduction, and the archive retains raw measurements for final selection. The selected matrix passes to subsequent BO, which starts with new observations.

A.1 THE SEARCH SCORE

For generation g, let $Z_g = \{z_{g1}, \dots, z_{gm}\}$ be a scrambled Sobol design in $[-s, s]^d$. Every population member uses these same latent points. A new design is generated for the next generation, so retained elites are evaluated again. For a candidate B, let $y_j = f(\phi_B(z_{gj}))$ and sort these responses as $y_{(1)} \le \cdots \le y_{(m)}$. With $1 \le k \le m$,

$$T_k(B) = \frac{1}{k} \sum_{j=1}^{k} y_{(j)}.$$

This score averages the lowest sampled objective values. It uses no surrogate, and the sample points are chosen before their objective values are observed.

For the clipping penalty, set $a_j = Bz_{gj}$ and $\widehat{a}_j = \mathrm{clip}(a_j, -\mathbf{1}, \mathbf{1})$. Then

$$C(B) = \frac{1}{mD} \sum_{j=1}^{m} \sum_{i=1}^{D} \left(\mathbf{1}\{|a_{ji}| > 1\} + |a_{ji} - \widehat{a}_{ji}| \right). \tag{4}$$

The first term is the fraction of coordinates outside the box, and the second is their mean absolute clipping displacement, averaged over all coordinates. This is the penalty used for selection. We report the directional-loss quantity in Eq. (10) separately as a geometric diagnostic.

For a scalar component v measured over a comparison set $\mathcal{S}$, define

$$\widetilde{v}_i = \frac{v_i - \min_{j \in \mathcal{S}} v_j}{\max_{j \in \mathcal{S}} v_j - \min_{j \in \mathcal{S}} v_j + \epsilon}, \qquad \epsilon > 0. \tag{5}$$

A constant component contributes zero. We combine the normalized T_k and C using Eq. (2). During reproduction, $\mathcal{S}$ is the current population. At final selection, it contains every evaluated matrix and its measurements from that evaluation. A matrix retained in several generations therefore has several separate entries, each based on that generation's sample design.

At final selection we use the raw (T_k, C) measurements, recompute their ranges over the full archive, and form a new score for every entry. This places entries on a common numerical scale while preserving each separate evaluation. A shorter search budget uses only measurements collected up to that budget. Exact archive-score ties are broken by earlier generation and then population order.

The regularizer prevents division by zero. Normalization can still magnify differences when a component has a small empirical range. A repeatedly retained matrix also gets several chances to obtain a favorable sampled score, so selecting its best entry can be optimistic even without observation noise. Independent rescoring would assess how well this score generalizes to new points. Our subsequent BO experiment evaluates the complete selection policy.

Reevaluating elites trades some new matrices for additional measurements of retained ones. The random-pool comparison includes both proposal adaptation and this allocation difference. The shuffled-score control tests the association with the complete score. Separating the objective and clipping components would require an additional ablation.

A.2 Evolutionary operators

Initialization draws independent matrices with entries from $\mathcal{N}(0, 1)$ and takes the reduced QR factor Q as B, retaining the signs returned by numerical QR. This convention leaves the deterministic identities in Appendix B unchanged.

At each reproduction step, the n_e lowest-scoring matrices are copied without modification. For each new child, two tournaments are drawn independently. A tournament samples $\min(\tau, N)$ distinct population members uniformly and selects the one with the lowest score. If tournament scores tie, the first tied candidate in the sampled order is selected. Elite-score ties preserve population order.

With probability p_c, crossover chooses each column independently from either parent with probability $1/2$. The resulting matrix is repaired by reduced QR. Otherwise the first parent is copied. Mutation then acts independently on each column: with probability p_m, it adds a Gaussian vector with covariance $\sigma^2 I_D$. The complete child is again repaired by reduced QR, even when no column was perturbed. Thus a child produced by crossover undergoes QR both before and after mutation. Mixing columns can change rank and scale, so applying these two repairs in sequence is part of the operator, rather than interchangeable with one final orthonormalization. If numerical repair fails, a new Gaussian matrix is drawn and orthonormalized.

Repair rejects nonfinite inputs or column norms at most 10^{-12}. After QR it checks rank with tolerance 10^{-8} and orthonormality with absolute and relative tolerances 10^{-6}. The input rank is not separately tested: dependent nonzero columns can receive an orthonormal completion from QR and be accepted. This convention, as well as the absence of sign canonicalization, is part of the evaluated operator.

QR also depends on column order: an earlier column can change the directions used to repair later columns. We use NumPy's reduced QR. Because a bounded region depends on its basis, changing the repair can change the search itself. A comparison with symmetric polar repair would help assess this design choice. Such a comparison would need an explicit rule for rank-deficient inputs.

All decisions use scores for minimization. Elites and offspring are evaluated on the next generation's sample points, with every reevaluation included in the embedding-search budget.

A.3 Subsequent BO and computational cost

The selected matrix is held fixed throughout subsequent BO. Its initial n_0 points are sampled uniformly in the latent box. Latent coordinates are rescaled to $[0, 1]^d$ for GP fitting, and the acquisition is optimized in those coordinates before mapping back through Eq. (1). The GP uses only observations collected during subsequent BO. This separates the quality of the selected embedding from the effect of inherited observations. Appendix G examines reuse in a separate historical study. The implementation checks for duplicate latent or mapped points and uses a new random candidate when acquisition optimization fails to return a valid point. These evaluations all count toward the BO budget T.

Embedding search uses exactly GNm objective calls, including reevaluated elites. Subsequent BO uses T calls including initialization. In addition to objective evaluation time, multiplying m latent points by a dense $D \times d$ matrix costs $O(mDd)$ per candidate, and a QR repair costs $O(Dd^2)$. Sorting responses for the top-k score costs $O(m \log m)$ per candidate. The numerical implementation also checks rank and orthonormality whenever it maps a point, adding $O(mDd^2)$ per candidate. Including tournaments and final archive sorting gives an embedding-search overhead bound of

$$O\big(GN\{mDd^2 + m\log m + \tau + \log(GN)\}\big).$$

The repeated rank and orthonormality checks are defensive implementation overhead. Validating a fixed candidate once would cost $O(Dd^2)$, followed by $O(mDd)$ for mapping its points. This engineering change could reduce overhead while preserving the objective-call count. Its runtime benefit remains to be measured. An archive storing every evaluated matrix and its scalar score components uses $O(GNDd)$ memory. Objective histories, if retained separately, require additional storage.

For BO, dense GP factorization with n observations costs $O(n^3)$ per model evaluation and $O(n^2)$ memory. Repeated model fitting and acquisition optimization add work whose actual cost depends on their numerical iteration counts. Objective-call counts and optimizer runtime therefore describe different resources, both of which matter when assessing end-to-end efficiency.

B Why the embedding changes the search region

This section explains the geometric properties used in Section 3. We first show why the basis matters and describe reachability and clipping. We then distinguish restriction of the search region from incomplete optimization within it. The results characterize the specified map and objective. Convergence of the full EES procedure remains an open question.

Proposition 1 (A bounded embedding depends on its basis)**.** *For $B^\top B = I_d$, $s > 0$, and an orthogonal matrix $Q \in \mathbb{R}^{d\times d}$,*

$$BQ\mathcal{Z}_s = B\mathcal{Z}_s \quad \Longleftrightarrow \quad Q \text{ is a signed permutation matrix.}$$

Signed permutations also preserve the clipped image $\phi_B(\mathcal{Z}_s)$. Other rotations can change it.

B.1 Proof of Proposition 1

Let $\mathcal{Z}_s = [-s, s]^d$. Since $B^\top B = I_d$, multiplying a set in the column space of B by $B^\top$ recovers its latent coordinates. Therefore

$$BQ\mathcal{Z}_s = B\mathcal{Z}_s \quad \Longleftrightarrow \quad Q\mathcal{Z}_s = \mathcal{Z}_s.$$

It remains to identify the orthogonal matrices that preserve the latent box.

Write $q_i^\top$ for row i of Q. Maximizing one output coordinate over the box gives

$$\max_{z\in\mathcal{Z}_s} |q_i^\top z| = s\|q_i\|_1.$$

The upper bound follows from the triangle inequality, and is attained by choosing each z_j to have magnitude s and the sign of q_{ij}. If $Q\mathcal{Z}_s = \mathcal{Z}_s$, the maximum must equal s, hence $\|q_i\|_1 = 1$. Orthogonality also gives $\|q_i\|_2 = 1$. Now

$$\|q_i\|_1^2 = \|q_i\|_2^2 + 2\sum_{j<k} |q_{ij}q_{ik}|.$$

Equality of the two norms forces every cross-product to vanish. Each row consequently has exactly one nonzero entry, of magnitude one. Orthogonal rows must place these entries in different columns, so Q is a signed permutation. Conversely, permuting coordinates or changing their signs preserves $\mathcal{Z}_s$, proving the equivalence.

For a signed permutation P, applying the same coordinatewise clipping to equal raw images gives

$$\phi_{BP}(\mathcal{Z}_s) = \phi_B(P\mathcal{Z}_s) = \phi_B(\mathcal{Z}_s).$$

To see that another rotation can change the clipped image, take $B = I_2$, $s = 1$, and let Q rotate by $45°$. The point $(1, 1)$ belongs to $\phi_B(\mathcal{Z}_1)$. To obtain it after clipping Qz, both raw coordinates would need to be at least one. Their sum, however, is $\sqrt{2}z_1 \leq \sqrt{2} < 2$. Thus $(1, 1)$ is not in $\phi_Q(\mathcal{Z}_1)$. □

The equivalence in the proposition concerns the image before clipping. After clipping, two different raw images can coincide: with $B = I_2$ and $s = 2$, both the unrotated square and its $45°$ rotation contain $[-1, 1]^2$, and both clip to that entire box. Also, equal reachable sets do not imply equal search scores on a finite design. A column permutation or sign change generally moves the Sobol points within the same set.

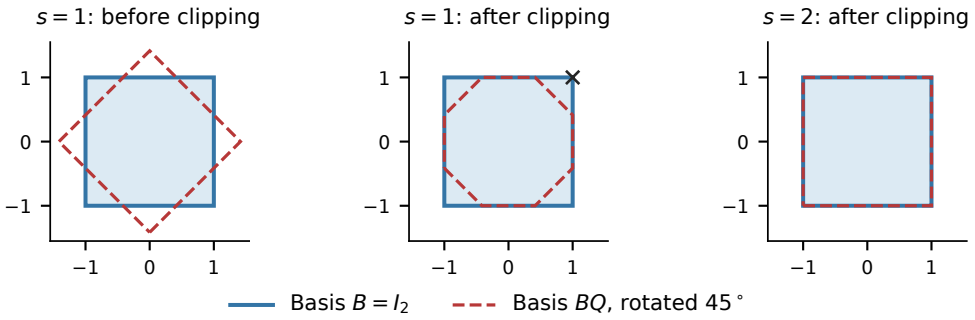

Figure 6: Exact two-dimensional examples of Proposition 1. Both bases span $\mathbb{R}^2$. At $s = 1$, rotation changes the raw region and the clipped region: the marked corner $(1, 1)$ is absent from the rotated image. At $s = 2$, the clipped images coincide despite different raw regions. The proposition's equivalence therefore applies before clipping.

B.2 Which points can a bounded embedding reach?

A search restricted to an embedding can only evaluate points in its image. The following test describes that image for a general rectangular domain, including targets on its boundary. It also makes explicit that normalization to $[-1, 1]^D$ does not restrict the original domain.

Let $\mathcal{X} = \prod_{i=1}^{D}[\ell_i, u_i]$, with $\ell_i < u_i$, and write

$$c_i = (u_i + \ell_i)/2, \qquad h_i = (u_i - \ell_i)/2, \qquad H = \operatorname{diag}(h_1, \ldots, h_D).$$

The physical-coordinate map is

$$\Phi_B(z) = c + H \operatorname{clip}(Bz, -\mathbf{1}, \mathbf{1}), \qquad z \in [-s, s]^d.$$

For a target $x \in \mathcal{X}$, define its normalized coordinates $v = H^{-1}(x - c)$ and the sets

$$I_0 = \{i : \ell_i < x_i < u_i\}, \qquad I_+ = \{i : x_i = u_i\}, \qquad I_- = \{i : x_i = \ell_i\}.$$

Proposition 2 (Exact reachability in a rectangular domain)**.** *The target x belongs to $\Phi_B([-s, s]^d)$ if and only if the following linear system is feasible:*

$$-s\mathbf{1} \leq z \leq s\mathbf{1}, \qquad B_{I_0} z = v_{I_0}, \qquad B_{I_+} z \geq \mathbf{1}, \qquad B_{I_-} z \leq -\mathbf{1}. \tag{6}$$

If x is interior and $B^\top B = I_d$, this is equivalent to $v \in \operatorname{col}(B)$ and $\|B^\top v\|_\infty \leq s$.

Proof. The affine transformation $x \mapsto v$ is a bijection between $\mathcal{X}$ and $[-1, 1]^D$. Hence $\Phi_B(z) = x$ exactly when $\operatorname{clip}(Bz, -\mathbf{1}, \mathbf{1}) = v$. For an interior coordinate, clipping returns $v_i \in (-1, 1)$ only when $(Bz)_i = v_i$. For an upper-bound coordinate, it returns one exactly when $(Bz)_i \geq 1$. The corresponding lower-bound condition is $(Bz)_i \leq -1$. These conditions are necessary for every preimage z. Conversely, a latent point satisfying all of them maps to v in every coordinate and therefore to x in physical coordinates. Adding the latent bounds proves both directions of Eq. (6).

For an interior target all conditions are equalities, $Bz = v$. A solution exists only if v lies in the column space. In that case orthonormality makes it unique: multiplication by $B^\top$ gives $z = B^\top v$,

while $BB^\top v = v$ verifies that this value solves the equation. It remains only to check its latent bounds. □

This proposition tests geometric feasibility without evaluating f. Finding a feasible target by BO is a separate optimization problem, and locating an unknown application's optimum requires information about its objective.

B.3 Coverage of an effective subspace

For the remainder of this section, x and f use the normalized coordinates of Eq. (1).

For an objective that depends on linear combinations $f(x) = g(A^\top x)$, where $A \in \mathbb{R}^{D\times d_e}$ has orthonormal columns, coverage can instead be considered in the d_e effective coordinates. Assume here that the whole latent box is free of clipping. Setting $W = A^\top B$, the accessible effective coordinates form $\mathcal{K} = W[-s, s]^d$, a linear image of a box, also called a zonotope. Its support in a direction u is

$$h_{\mathcal{K}}(u) = \max_{y\in\mathcal{K}} u^\top y = \max_{\|z\|_\infty \le s} u^\top W z = s\|W^\top u\|_1.$$

The last equality maximizes each coordinate separately. A centered radius-r ball has support r in every unit direction. By the standard support criterion for convex containment, the largest such ball inside $\mathcal{K}$ therefore has radius

$$r_{\text{in}}(\mathcal{K}) = s \min_{\|u\|_2=1} \|W^\top u\|_1 \ge s \min_{\|u\|_2=1} \|W^\top u\|_2 = s\sqrt{\lambda_{\min}(WW^\top)}. \tag{7}$$

The inequality uses $\|w\|_1 \ge \|w\|_2$, and the final equality is the Rayleigh-quotient formula. This expression remains valid when W lacks full row rank: both the guaranteed radius and the full-dimensional inradius are then zero. In particular, fewer than d_e latent coordinates cannot cover a ball in $\mathbb{R}^{d_e}$.

When W has full row rank, its smallest singular value supplies a positive coverage bound. Thus both alignment with the effective directions and the size of the latent box matter. This conditional argument motivates learning B when the objective has the stated low-dimensional form. EES itself can be applied without knowing A.

In particular, the Hartmann experiment uses $d = 5 < d_e = 6$, so its effective-coordinate inradius is zero. Its empirical improvement concerns finding useful points in a lower-dimensional image. Explaining that result requires more than full-dimensional coverage.

B.4 When does clipping change local distances?

Clipping can map distinct latent points to the same point in the original space. Beyond counting clipped coordinates, we can ask which latent directions lose information. The next result supplies the directional diagnostic used in the application analysis.

Write $r_i^\top$ for row i of B and $\omega_i = \|r_i\|_2^2$. Orthonormal columns give

$$\sum_{i=1}^{D} r_i r_i^\top = I_d, \qquad \sum_{i=1}^{D} \omega_i = d. \tag{8}$$

The first identity is simply the row expansion of $B^\top B$. The second follows by taking its trace. Thus the rows together preserve latent squared lengths before clipping.

Proposition 3 (Clipping-free domain and local distance loss)**.** *The map ϕ_B is clipping free on all of $[-s, s]^d$ if and only if*

$$s \max_i \|r_i\|_1 \le 1. \tag{9}$$

Away from points satisfying $|r_i^\top z| = 1$, let $S(z) = \{i : |r_i^\top z| > 1\}$ and $J_B(z)$ be the Jacobian of ϕ_B. Then

$$\begin{aligned} J_B(z)^\top J_B(z) &= I_d - \sum_{i\in S(z)} r_i r_i^\top, \\ \sigma_{\min}(J_B(z))^2 &= 1 - \lambda_{\max}\left(\sum_{i\in S(z)} r_i r_i^\top\right). \end{aligned} \tag{10}$$

Proof. For any row, the triangle inequality gives $|r_i^\top z| \le s\|r_i\|_1$ throughout the latent box. Choosing $z_j = s\,\mathrm{sign}(r_{ij})$ attains this upper bound. Every coordinate therefore stays within its normalized bounds exactly when the largest of these rowwise maxima is at most one. This proves Eq. (9), including necessity.

To obtain the local metric, consider a point away from a clipping breakpoint. An unclipped output coordinate changes by $r_i^\top \Delta z$ under a small latent displacement. A clipped coordinate is locally constant and has derivative zero. If D_U is the diagonal indicator of the unclipped coordinates, these derivatives give

$$J_B(z) = D_U B, \qquad J_B(z)^\top J_B(z) = B^\top D_U B = B^\top B - \sum_{i \in S(z)} r_i r_i^\top.$$

Substitution of $B^\top B = I_d$ proves the first identity. The matrix $M_S = \sum_{i\in S(z)} r_i r_i^\top$ is positive semidefinite, and Eq. (8) gives $M_S \preceq I_d$. Thus its eigenvalues lie between zero and one. The smallest eigenvalue of $I_d - M_S$ is $1 - \lambda_{\max}(M_S)$. Since the eigenvalues of $J_B^\top J_B$ are the squared singular values of J_B, this proves the second identity. □

For a unit latent direction u, the retained squared length is $\|J_B(z)u\|_2^2 = 1 - u^\top M_S u$. Maximizing the loss over u gives $\lambda_{\max}(M_S)$. A value close to one means that some latent direction is nearly lost, even if few original coordinates clip. The effect depends on the lengths and alignment of the clipped rows, which explains why clipping frequency alone does not determine local conditioning. The objective's sensitivity to each direction is a separate property.

Two related bounds help interpret changes in latent scale. The largest centered ball in unrestricted latent space on which clipping is absent has radius

$$R_{\mathrm{ball}} = \frac{1}{\max_i \|r_i\|_2}, \qquad 1 \le R_{\mathrm{ball}} \le \sqrt{D/d}. \tag{11}$$

Indeed, maximization over a radius-R ball gives $\max_{\|z\|_2 \le R} |r_i^\top z| = R\|r_i\|_2$ by Cauchy–Schwarz, with equality in the row direction. Also $\omega_i \le 1$ because ω_i is a diagonal entry of the orthogonal projector $BB^\top$, while $\max_i \omega_i \ge d/D$ by Eq. (8). These give the two radius bounds. If the ball must also lie inside $[-s, s]^d$, its largest allowable radius is $\min(s, R_{\mathrm{ball}})$.

Finally, clipping cannot increase the norm from the origin, so

$$\|\phi_B(z)\|_2 \le \min\{\|Bz\|_2, \sqrt{D}\} = \min\{\|z\|_2, \sqrt{D}\} \le \min\{s\sqrt{d}, \sqrt{D}\}.$$

Consequently $s\sqrt{d}$ is the maximum raw radius before clipping and an upper bound on the reachable radius afterward. Equal values of $s\sqrt{d}$ can still give different coverage, clipping, and BO behavior.

B.5 Latent size, coverage, and optimization difficulty

For fixed B and $s_1 \le s_2$, the latent boxes are nested, hence so are their clipped images. The best attainable objective value cannot worsen when s increases. BO then searches a larger latent domain with the available evaluations, so its achieved value can improve or worsen. Separately learning B at each setting removes the nesting guarantee. Independently constructed matrices with different numbers of columns can also describe nonnested regions.

For a uniformly distributed $z \in [-s, s]^d$, zero mean and covariance $(s^2/3)I_d$ give

$$\mathbb{E}\|Bz\|_2^2 = \frac{ds^2}{3}, \qquad \frac{1}{D}\sum_{i=1}^{D} \mathrm{Var}((Bz)_i) = \frac{ds^2}{3D}. \tag{12}$$

The first equality uses $B^\top B = I_d$ and the second takes the trace of $\mathrm{Cov}(Bz) = (s^2/3)BB^\top$. At $(s, d) = (3, 10)$ the mean pre-clipping coordinate variance is 0.5 for Rover's $D = 60$ and approximately 0.0773 for SVR's $D = 388$. Thus a shared numerical pair can produce different spread and clipping behavior in the original space. These identities use normalized coordinates. Physical units enter through the affine domain transformation.

The application grid provides task-specific observations at a fixed BO horizon. Selecting (s, d) automatically remains an empirical design problem. It requires charged pilot evaluations and independent assessment, with a common scoring scale across configurations. Increasing d to improve representation also gives BO more coordinates to model and search.

B.6 Separating region quality from remaining optimization error

A nearly flat curve need not mean that BO has found the original optimum. It may instead be approaching the best value available in its selected region. For a continuous deterministic objective on the compact domain, define

$$f^*_{B,s} = \min_{z\in[-s,s]^d} f(\phi_B(z)), \qquad f^* = \min_{x\in\mathcal{X}} f(x).$$

For the best point $x_t^{\min}$ evaluated during subsequent BO in this region,

$$\underbrace{f(x_t^{\min}) - f^*}_{\text{total regret}} = \underbrace{f^*_{B,s} - f^*}_{\text{restriction error}} + \underbrace{f(x_t^{\min}) - f^*_{B,s}}_{\text{remaining optimization error}} . \tag{13}$$

Both terms are nonnegative. The first is fixed once the embedding is selected. The second may decrease with further BO evaluations. Using this identity as a diagnostic requires computing or bounding the best attainable value in the region. For a physical domain, replace ϕ_B by its affine unnormalization Φ_B.

The shifted Sphere used here provides an exact diagnostic without running an auxiliary optimizer.

Proposition 4 (Exact restriction error for the shifted Sphere)**.** *Let $f(x) = \|x - 5a\|_2^2$ on $[-5,5]^D$, where $\|a\|_2 = 1/2$, and let $\Phi_B(z) = 5\,\mathrm{clip}(Bz, -1, 1)$ with $B^\top B = I_d$. For $s \geq 1/2$,*

$$f^*_{B,s} = 25\|(I_D - BB^\top)a\|_2^2, \qquad z^* = B^\top a \text{ is a minimizer.} \tag{14}$$

Proof. Since $\|B^\top a\|_\infty \leq \|a\|_2 = 1/2$, z^* satisfies the latent bounds. Its raw image $BB^\top a$ has norm at most $1/2$ and is not clipped. Orthogonal projection minimizes squared distance over the whole column space, attaining Eq. (14). Every unclipped candidate has at least this value. Any clipped candidate has an output coordinate of magnitude one in normalized units. Since $|a_i| \leq 1/2$, that coordinate alone contributes at least $25/4$ to the objective. The projection candidate has value at most $25\|a\|_2^2 = 25/4$. Clipping therefore cannot improve upon it. □

This result applies to the actual shifted-Sphere construction and all latent bounds used here. Its target-norm and quadratic assumptions are essential. It also separates subspace alignment from basis orientation: the Sphere restriction error in this regime is unchanged by replacing B with BQ, even though finite-design scores and BO trajectories may change.

For the other deterministic functions, numerical minimization within an embedding supplies only a best-known feasible value $\widehat{f}_{B,s} \geq f^*_{B,s}$ unless accompanied by a certificate. Substituting it into Eq. (13) overestimates restriction error and underestimates remaining optimization error. A later BO point may beat that numerical reference, which should therefore be labeled as an upper bound on the attainable minimum. Matrix-level restriction-error measurements are left for future analysis. The identity and the Sphere formula specify useful diagnostics for interpreting longer runs.

Evaluating new sample points answers a complementary question: whether an embedding still gives low objective values beyond those used to select it. Repeating the original top-five statistic on these points tests the quality of the sampled region. Its association with subsequent BO could guide more focused tests of the selection mechanism. For noisy Rover evaluations, analyzing the expected objective additionally requires independent repeated evaluation of returned points. Raw observed minima describe the noisy responses.

C Experimental details

This section defines the test problems and the quantities shown in the figures. The synthetic experiments test embedding selection under known transformations. The applications test whether a useful region can be found when no low-dimensional generative model is supplied to EES.

C.1 Synthetic objectives and transformations

Unshifted test functions often place an optimum at the center of the domain. Every linear embedding through that center then has access to the same favorable point, which can conceal differences in

embedding quality. We therefore move a known optimum away from the center. Random rotations distribute the active variables across the original coordinates. The transformed functions are shared by all methods within a comparison and are constructed without using their optimization results.

Write $A \in \mathbb{R}^{D \times d_e}$ for an orthonormal active matrix and $a \in \mathbb{R}^{d_e}$ for a target vector of norm 0.5. Here d_e is the effective dimension of the synthetic objective. The latent dimension d used by EES is chosen separately. The following definitions use the original input coordinates x. All functions except Sphere have domain $[-1, 1]^D$.

- **Ackley:** $f(x) = \mathrm{Ackley}(20(A^\top x - a))$, with $d_e = 5$ and optimum zero.
- **Rosenbrock:** $f(x) = \mathrm{Rosenbrock}(\mathbf{1} + c(A^\top x - a))$, with $d_e = 5$, optimum zero, and $c = 1$ or 5. The two variants share the shift and differ in active-coordinate scale.
- **Branin and Hartmann:** let g denote the conventional function on its box $[\ell, u]$. We use $f(x) = g(\ell + (u - \ell) \odot \mathrm{sigmoid}(3(A^\top x - b)))$, with $d_e = 2$ for Branin and 6 for Hartmann. The sigmoid acts coordinatewise. The offset b is chosen so that a designated optimum of g occurs at an active target of norm 0.5. This smooth map keeps every evaluation in the canonical function's domain.
- **Sphere:** on $[-5, 5]^D$, $f(x) = \|x - 5a\|_2^2$, where now $a \in \mathbb{R}^D$ has norm 0.5. Sphere provides a full-dimensional quadratic control.

For the embedded functions, a representative of the designated optimum is checked to lie inside the input box, with maximum absolute coordinate at most 0.85. Thus the reference optimum is attainable in the original domain, although it need not be attainable through any particular EES embedding. The reference values are 0.3978873577 for Branin and -3.3223680114 for Hartmann. Ackley, Rosenbrock, and Sphere use zero.

The feedback-control study uses ten transformation roots, 410001–410010, and five optimizer seeds, 0–4. The synthetic method comparison uses three different roots, 510001–510003, and three optimizer seeds. These studies use the same function families with different realized transformations. The dimension study also generates separate transformed instances at each D, so its comparisons include variation between instances.

Why these functions? The base families are standard optimization tests (Surjanovic & Bingham, 2013). Embedded Branin and Hartmann also appear in high-dimensional BO evaluations (Letham et al., 2020; Papenmeier et al., 2022). Their roles differ. Ackley tests search among many local minima. Branin and Hartmann test low-dimensional objectives with multiple basins, at effective dimensions two and six. The two Rosenbrock variants test scale sensitivity within the same curved-valley family. Sphere removes multimodality and the low-effective-dimension assumption, providing a quadratic control with the exact embedding restriction error in Proposition 4. Sigmoid warping changes the canonical functions, and the bounded transformed domain can exclude some canonical basins. These constructions probe several distinct difficulties. Extending the suite to nonsmooth and weakly structured objectives would broaden the assessment.

C.2 Initialization and baseline implementations

Subsequent BO fits a GP to new observations and selects evaluations with LogEI (Ament et al., 2023). Its five initial evaluations are included in the budget T in Table 1. The embedding controls use the same optimizer, initial sample count, and paired random seeds. A shared latent seed yields different original-space inputs when the selected embeddings differ.

The GP normalizes latent inputs to $[0, 1]^d$ and standardizes responses. It uses the BoTorch 0.17.2 single-task defaults: a learned constant mean, an ARD radial-basis covariance without a separate output scale, and learned Gaussian noise variance. Each length scale has prior $\mathrm{LogNormal}(\sqrt{2} + \frac{1}{2}\log d, \sqrt{3})$ and lower bound 0.025. The noise variance has prior $\mathrm{LogNormal}(-4, 1)$ and lower bound 10^{-4} on the standardized-response scale. The two prior arguments are the underlying normal location and standard deviation. Length scales and noise start at their prior modes and are optimized directly subject to these bounds.

We fit the exact marginal likelihood after each observation and optimize analytic minimization LogEI for one new point, using ten restarts and 128 raw candidates. Fitting is attempted up to three

Table 3: Initial evaluations in the synthetic comparison at $D = 50$. All are included in the optimization budget T in Table 1. Sobol samples throughout the run without a separate fitting phase.

Method	Initial calls	Source and construction
EES and controls	5	Shared subsequent GP–LogEI optimizer
BAxUS	10	Adaptive nested embeddings (Papenmeier et al., 2022)
TuRBO-1	10	Single trust-region variant (Eriksson et al., 2019)
DSEBO	1	Authors' implementation (Qian et al., 2026)
Vanilla BO	22	Dimension-scaled prior, $\lceil 3\sqrt{D} \rceil$ initial points (Hvarfner et al., 2024)
Spherical linear BO	30	Spherical linear kernel (Doumont et al., 2026)
MSR	21	Authors' implementation (Papenmeier et al., 2025)
Standard Matérn	20	Authors' implementation (Xu et al., 2025)
CMA-BO	20	Covariance-adaptation BO (Ngo et al., 2024)

times before a uniform latent proposal is used. Acquisition failures use the same fallback. The retry schedule does not specify different double-precision jitters. Recorded environments use GPyTorch 1.15.2 and Torch 2.11.0 with BoTorch 0.17.2. Earlier and application workers differ in NumPy and SciPy versions, and the records do not specify every transitive dependency.

The direct-method comparison retains each method's initialization rule, as listed in Table 3. The resulting curves compare complete methods, including the effects of their different surrogates and initial designs on early progress.

DSEBO, MSR, and standard Matérn use the authors' implementations in the synthetic comparison. The application and dimension studies use a separate reproduction of DSEBO, whose results must be distinguished from those of the authors' implementation in Figure 3. The application panel contains six direct methods: Sobol, BAxUS, TuRBO-1, DSEBO, Vanilla BO, and spherical linear BO. Vanilla BO initializes with 24 points on Rover and 60 on SVR. Spherical linear BO uses 30 on both.

The evolutionary settings are fixed across the principal studies: four elites, tournament size three, crossover probability 0.8, per-column mutation probability 0.3, and Gaussian mutation standard deviation 0.1. The score-normalization constant is $\epsilon = 10^{-12}$. Table 1 gives the score and population settings. These studies vary search budget, latent bounds, or dimension while retaining the same evolutionary operators across objectives. Appendix G identifies the distinct protocol used in the earlier reuse study.

The application grid includes all twelve configurations and five seeds on both tasks. Interrupted runs were retried at the same settings, yielding 120 complete runs.

C.3 Metrics and uncertainty

For minimization, the best-so-far value (also called the incumbent) after t evaluations is $f_t^{\min} = \min_{j \le t} f(x_j)$. The synthetic curves plot

$$L_t = \log_{10}\big(\max\{f_t^{\min} - f^*, 0\} + \varepsilon_v\big)\,.$$

The logarithm makes improvements at different error scales visible. The offset ε_v prevents $\log(0)$ and is fixed across methods: 10^{-12} for Sphere, 10^{-10} for Ackley and Rosenbrock, 10^{-9} for Branin, and 10^{-8} for Hartmann. The maximum with zero also prevents tiny round-off errors in a known optimum from producing an invalid logarithm. These are fixed numerical safeguards.

For the controlled curves, we first average five seeds within each transformed instance, then average the ten instance means. At each BO evaluation, the shaded band is a Student-t 95% interval across these ten means. The interval is calculated separately at each evaluation. The paired final differences in Table 5 use a different calculation: 10,000 hierarchical-bootstrap samples resample transformed instances and paired seeds within each objective, then give the six objective means equal weight. Methods and search budgets use the same resampled indices. The displayed intervals are the 2.5th and 97.5th percentiles for each contrast separately. Selecting a search budget requires the simultaneous calculation below.

For simultaneous upper bounds, let $\widehat{\Delta}_j$ denote an observed mean difference and Δ^*_{bj} its value in bootstrap draw b. Over the eight combinations of two embedding controls and four search budgets, the calculation is

$$c_{.95} = \text{quantile}_{.95}\left\{\max_{j=1,\ldots,8}(\Delta^*_{bj} - \widehat{\Delta}_j)\right\}_{b=1}^{10000}, \qquad U_j = \widehat{\Delta}_j + c_{.95}.$$

Table 6 reports U_j. These approximate bootstrap-based bounds describe uncertainty in the fixed six-objective suite. Generalization to other problem classes requires broader evaluation. We report the bounds directly, avoiding interpretation of the fraction of uncentered bootstrap draws above zero as an exact null-test p-value.

The synthetic method-comparison and dimension-study curves first average the three seeds in each transformed instance, then average the three instance means. Bands show one standard error across these instance means. With only three instances, the uncertainty estimates are necessarily imprecise. The principal application curves average five seeds for a fixed objective, and the reduced-budget follow-up averages ten. Both show one standard error. These bands describe the observed settings without adjusting for application-configuration selection. A paired comparison requires the differences between matched runs, beyond visual inspection of overlapping bands. For applications, we report the final value and the curve mean, $T^{-1}\sum_{t=1}^{T} f_t^{\min}$. The curve mean averages the best-so-far values and rewards improvements found earlier in the run. All initial evaluations count toward both metrics.

C.4 Evaluation cost and computation

Each evaluated embedding costs m objective calls, including a retained elite when it is evaluated on the next generation's design. Table 4 gives the complete accounting. At a chosen search budget, EES and both embedding controls have the same cost. The equal-budget Sobol reference spends this entire number of calls in the original input domain and retains its best observed value.

Table 4: Per-run objective evaluations. The final column is the budget of the corresponding direct Sobol reference.

Generations	Embedding search	Subsequent BO	Total
5	25,600	100	25,700
10	51,200	100	51,300
15	76,800	100	76,900
20	102,400	100	102,500

The controlled GP calculations use double precision. Experiments were run on heterogeneous workers, and the available records do not support a common-hardware runtime comparison across all methods. The reported comparisons therefore concern objective evaluations and subsequent BO quality. Sample points within a search generation can be evaluated independently. Measuring the resulting parallel speedup requires a separate common-hardware evaluation.

D Further results on embedding selection

The main controlled curves use the full twenty-generation budget. Here we show how the selection benefit develops across all tested lengths. Ten generations is the first tested length with an individual contrast interval below zero against the random pool. The simultaneous bounds account for all eight control-and-budget comparisons and give a more cautious assessment of this data-selected threshold.

At twenty generations, EES has a lower final value than the random pool in 64.0% of the 300 paired runs, and than shuffled scores in 70.0%. Table 6 supplies the simultaneous upper bounds. The ten-generation random-pool bound includes zero, even though its individual contrast interval is below zero. At twenty generations both simultaneous bounds are negative.

The mean advantage retains its sign under both alternative weightings (Table 7). Removing Hartmann reduces its magnitude by almost half, so this task contributes substantially to the aggregate even though it does not determine its sign alone.

Table 5: Final log-regret difference, EES minus comparator, with paired 95% hierarchical-bootstrap intervals. Negative values favor EES. All embedding methods use equal budgets. Sobol spends the same total budget directly in the original space.

G	Random pool	Shuffled scores	Equal-budget Sobol
5	-0.037 $[-0.127, 0.054]$	-0.150 $[-0.241, -0.060]$	-0.946 $[-1.035, -0.858]$
10	-0.114 $[-0.217, -0.017]$	-0.173 $[-0.275, -0.075]$	-0.886 $[-0.965, -0.806]$
15	-0.115 $[-0.218, -0.018]$	-0.186 $[-0.291, -0.083]$	-0.837 $[-0.924, -0.754]$
20	-0.136 $[-0.229, -0.043]$	-0.196 $[-0.290, -0.105]$	-0.787 $[-0.876, -0.699]$

Table 6: Simultaneous one-sided 95% upper bounds for mean EES-minus-control final log regret, across both controls and all four search budgets. Negative bounds favor EES. The construction is given in Appendix C.3.

G	Random pool	Shuffled scores
5	0.0812	-0.0317
10	0.0043	-0.0549
15	0.0037	-0.0680
20	-0.0170	-0.0776

Table 7: Sensitivity of the twenty-generation mean final log-regret contrast to fixed-suite weighting. The five-family mean first averages the two Rosenbrock scales. Omitting Hartmann leaves five equally weighted variants. These descriptive reweightings summarize the same recorded runs.

Weighting	EES minus random pool	EES minus shuffled scores
Six variants	-0.136	-0.196
Five function families	-0.147	-0.218
Without Hartmann	-0.073	-0.105

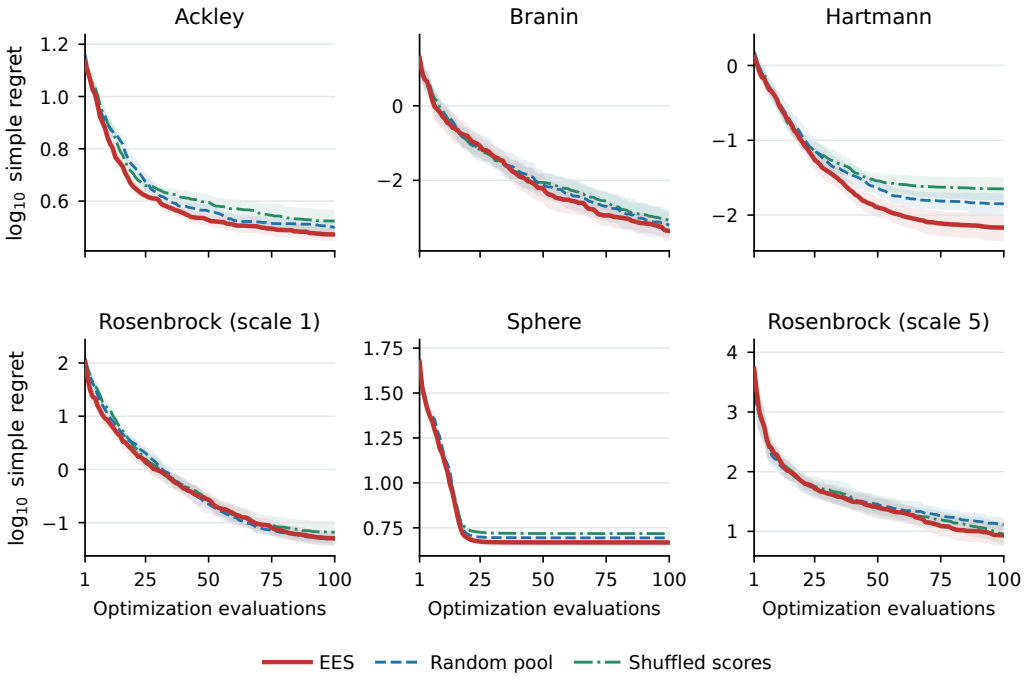

Figure 7: Subsequent BO after ten embedding-search generations. All three methods use the corresponding budget in Table 4. Lines average ten transformed-instance means, each based on five seeds. Shading gives a 95% Student-t interval calculated separately at each BO evaluation, as in Figure 2.

Figure 7 gives the three methods' trajectories after ten generations. Figure 8 instead holds EES fixed and compares the four search budgets. The latter shows why a single aggregate cannot describe the budget response: Hartmann benefits from more embedding search, while some curves cross on Branin and the smaller-scale Rosenbrock problem.

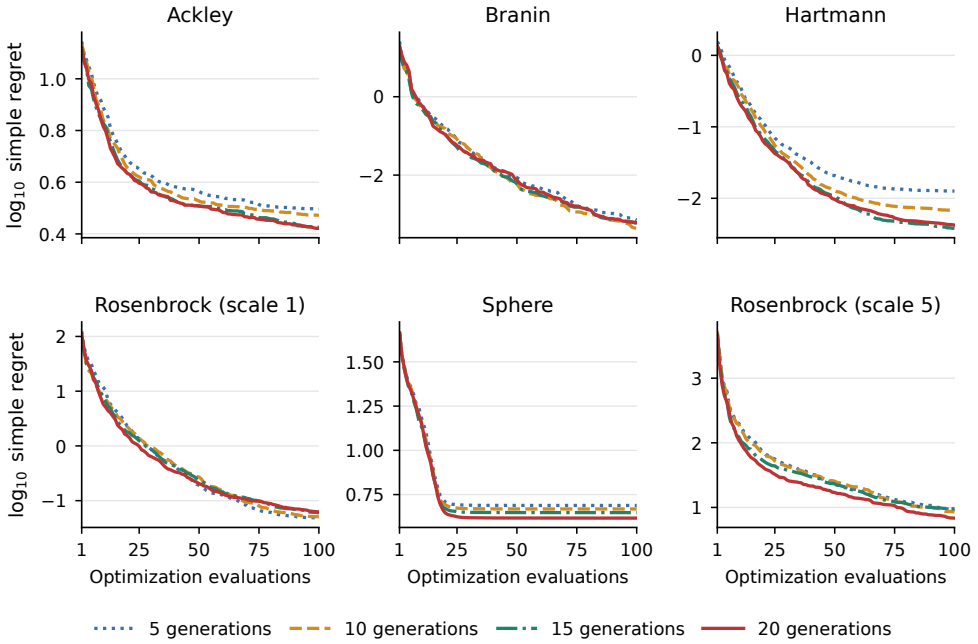

Figure 8: Subsequent BO for EES after different embedding-search budgets. Lines average ten transformed-instance means, each based on five seeds. The embedding-search budgets for $G = 5, 10, 15, 20$ are listed in Table 4. Only subsequent BO is displayed.

Table 8: Change in EES final log regret when embedding search is extended by five generations. The largest observed mean improvement occurs from five to ten generations. The intervals include zero in each case.

Generations	Mean change	Paired 95% interval
5 to 10	-0.0863	$[-0.1818, 0.0067]$
10 to 15	-0.0132	$[-0.1022, 0.0733]$
15 to 20	-0.0149	$[-0.0807, 0.0504]$

Figure 9 compares final values as a function of the full evaluation cost. It includes Sobol sampling in the original space as a total-budget reference alongside the embedding-selection controls. The task-specific curves show where additional embedding search produces useful gains and where the ordering changes with budget.

E Method comparison and dimension scaling

Figure 10 adds MSR, the standard Matérn GP, CMA-BO, and Sobol to the six methods emphasized in the main comparison. Every function and method is retained here. Table 10 supplies a compact summary. Ranks are computed within each objective–instance–seed combination and then averaged across all 54 combinations. They summarize relative ordering. The curves and endpoint differences show the magnitude of improvement, with EES using a larger total evaluation budget because of embedding search.

Table 11 makes the variation across transformed problems explicit. EES improves over the selected direct reference on every instance of Branin, Hartmann, and both Rosenbrock variants. Ackley fa-

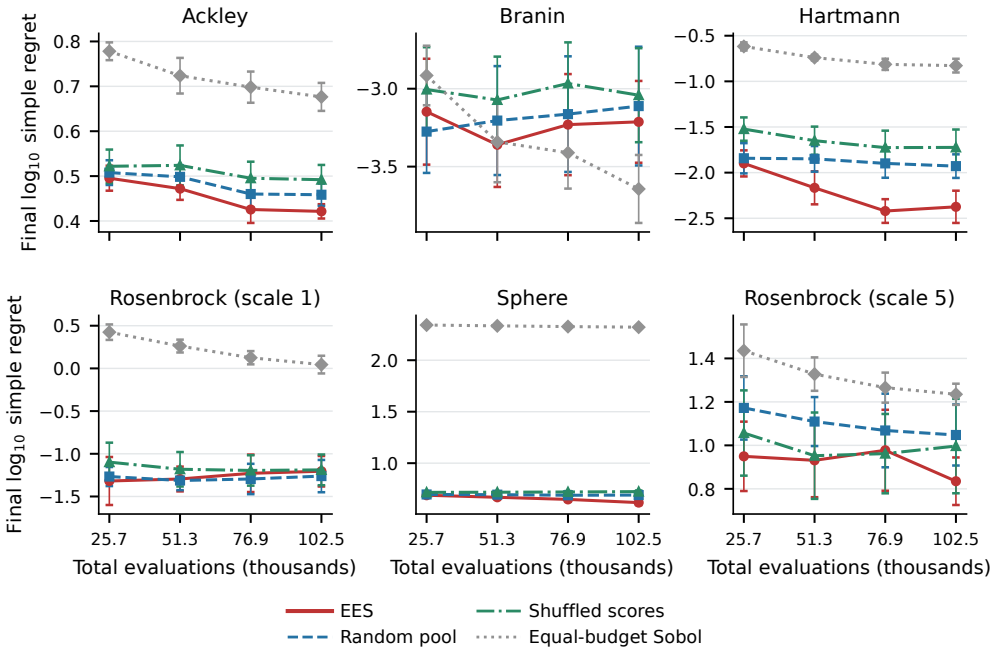

Figure 9: Final performance versus total objective evaluations. Embedding methods count embedding search and subsequent BO together, using the budgets in Table 4. Sobol uses the same total directly. Points average ten transformed-instance means, with Student-t 95% intervals. Lines connect the four measured budgets as visual guides.

Table 9: Final mean $\log_{10}$ regret over three instances and three seeds per instance. The direct reference is selected by its lowest observed mean on each objective. Bold marks the lower observed mean, without implying statistical significance. EES additionally uses the embedding-search budget in Table 1, so total budgets differ.

Objective	EES	Best direct mean	Direct method
Ackley	0.416	**0.371**	DSEBO
Branin	**−2.971**	−1.968	BAxUS
Hartmann	**−2.504**	−1.154	BAxUS
Rosenbrock (scale 1)	**−1.381**	−0.547	BAxUS
Sphere	**0.606**	0.620	BAxUS
Rosenbrock (scale 5)	**0.727**	1.207	BAxUS

Table 10: Descriptive mean paired final rank across the six synthetic objectives. Bold marks the lowest mean rank. EES uses the additional embedding-search budget in Table 1, so the ranks compare methods at unequal total cost.

Method	Mean rank
EES	**1.352**
BAxUS	3.056
DSEBO	3.630
Vanilla BO	5.491
Spherical linear BO	5.824
TuRBO-1	6.407
MSR	6.500
Standard Matérn	6.907
CMA-BO	7.519
Sobol	8.315

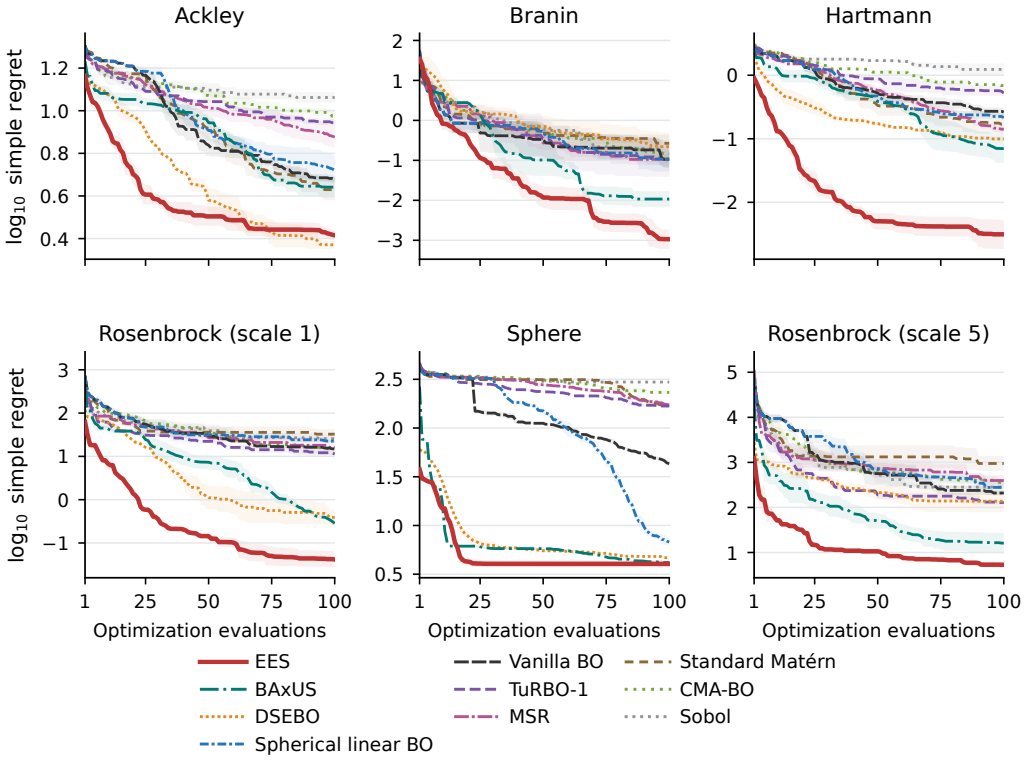

Figure 10: Complete synthetic comparison at $D = 50$. Lines average three transformed-instance means, each based on three seeds. Bands show one standard error across instances. Only subsequent BO is displayed for EES, which additionally uses the embedding-search budget in Table 1. Direct methods use only the displayed optimization budget.

vors DSEBO, while Sphere changes sign across instances. With only three instances, the individual effects are more informative than a pooled rank standard error.

Table 11: Paired final mean log-regret differences, EES minus direct method, after averaging the three seeds within each instance. These descriptive effects use a comparator selected by its mean across the same three instances. Budgets differ as in Figure 3.

Objective	Direct method	Instance 1	Instance 2	Instance 3
Ackley	DSEBO	0.035	0.042	0.056
Branin	BAxUS	−0.852	−1.116	−1.042
Hartmann	BAxUS	−1.632	−0.478	−1.937
Rosenbrock (scale 1)	BAxUS	−0.751	−0.582	−1.171
Sphere	BAxUS	−0.070	−0.010	0.039
Rosenbrock (scale 5)	BAxUS	−0.337	−0.283	−0.818

E.1 Increasing the original dimension

The dimension study uses embedded Ackley and Rosenbrock with active-coordinate scale 1 at $D = 50, 200, 500, 1000$, with three transformed instances and three seeds for each function. The latent dimension remains five. This tests how the embedding-selection procedure behaves when the original space grows while the subsequent BO dimension stays small. Each dimension uses independently generated transformed instances.

The mutation scale also stays fixed. Before QR repair, a perturbed column receives noise with expected squared norm $D\sigma^2$. Thus coverage, proposal size, and model-fitting difficulty can all

contribute to the dimension response. Testing dimension-scaled proposals would help separate these effects.

Figure 11 and Table 12 show favorable endpoints for EES in subsequent BO at $D = 200$. At $D = 500$ and $D = 1000$, BAxUS wins most pairs with fewer total evaluations. TuRBO-1 splits the pairs evenly at $D = 500$ and wins 17 of 18 at $D = 1000$. These patterns motivate adapting the latent dimension and matrix-search proposals as the original dimension grows.

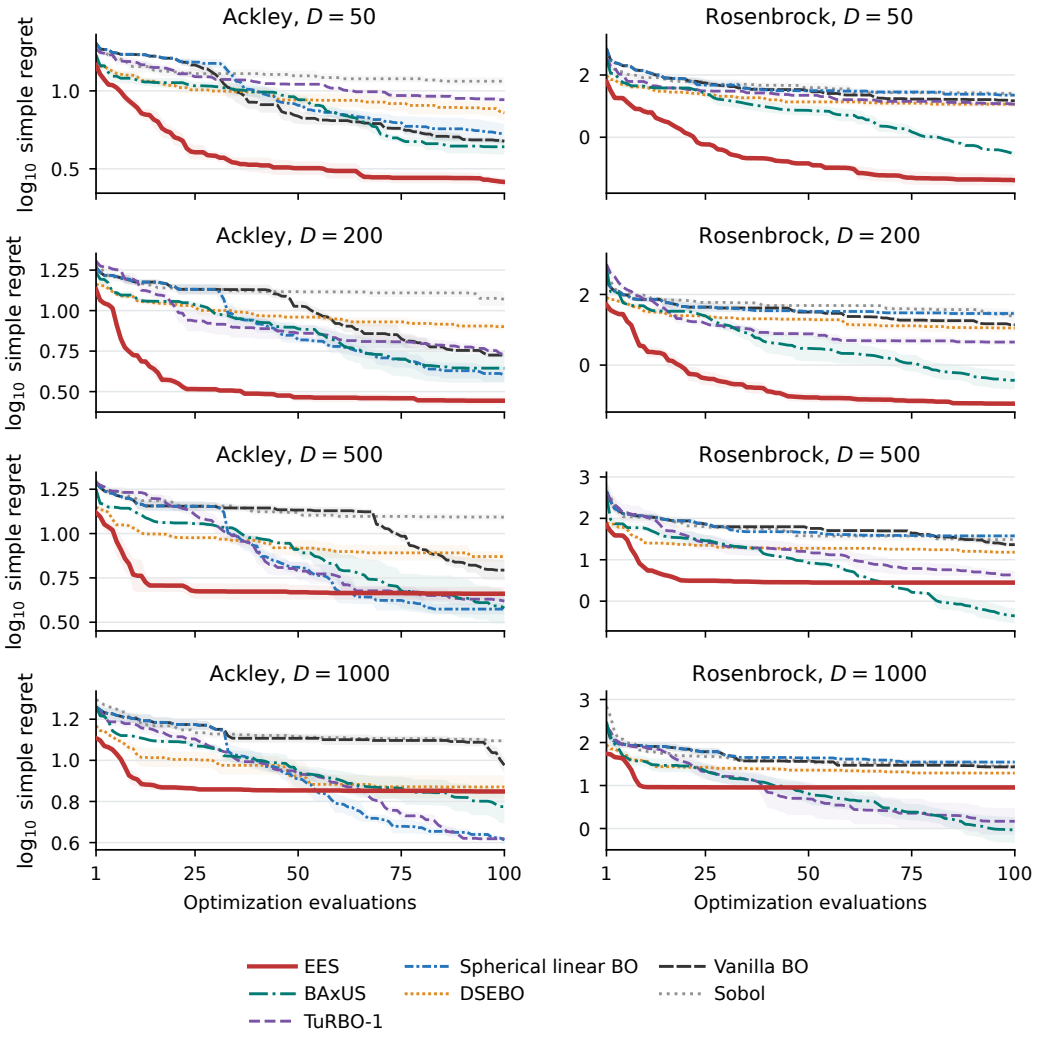

Figure 11: Optimization curves at increasing original dimension. Lines average three transformed-instance means, each based on three seeds. Bands show one standard error across instances. EES uses a five-dimensional latent domain for subsequent BO and additionally uses the full embedding-search budget in Table 4. Direct methods use only the displayed optimization budget. DSEBO here uses the reproduction described in Appendix C.2.

F Application definitions and additional results

F.1 Rover trajectory planning and support-vector regression

Rover follows the trajectory-design problem used by Wang et al. (2018). Thirty two-dimensional waypoints define a spline path, giving 60 optimization variables in $[0, 1]^{60}$. The objective integrates terrain cost along the path and penalizes discrepancies from the prescribed start and goal. The

Table 12: Number of paired final comparisons won by EES, out of 18 at each dimension. All methods receive the same displayed optimization budget, while EES additionally uses its full embedding-search budget (Table 4). DSEBO is the reproduction variant in this study.

D	BAxUS	TuRBO-1	Spherical linear BO	DSEBO	Vanilla BO	Sobol
50	18	18	18	18	18	18
200	14	18	17	18	18	18
500	5	9	10	17	17	18
1000	3	1	9	15	16	17

implementation subtracts a constant five from the cost, so a negative reported value corresponds to an unshifted cost below five. A seeded Gaussian waypoint perturbation of standard deviation 10^{-4} is applied during evaluation. This produces a noisy objective with repeatable seeded runs.

The regression benchmark follows the CT-slice task convention used in the spherical-linear BO implementation (Doumont et al., 2026), with the source data from Graf et al. (2011). The source contains 53,500 rows. We fix 500 selected rows, use 250 to fit the regressor and 250 to evaluate it, and optimize 385 input scale factors together with the SVR parameters C, γ, and ϵ. Retaining this small, fixed split preserves the benchmark objective across methods and runs. The loss is RMSE on the normalized target.

The fixed subset is drawn with NumPy's `RandomState(388)`. The first 250 selected rows form the training set and the remaining rows form the validation set. For an optimization vector $u \in [0,1]^{388}$, the input scale factors are $\ell_j = \exp(4u_j - 2)$ for $j = 1, \ldots, 385$. The regression parameters are

$$\epsilon = 0.01 \cdot 100^{u_{386}}, \qquad \gamma = 0.1 \cdot 30^{u_{387}}, \qquad C = 0.01 \cdot 500^{u_{388}}.$$

Two data conventions matter when interpreting this benchmark. The 385 input columns retain the source patient identifier alongside 384 image-derived attributes. Input and target min–max normalization is fitted on the full source data before selecting the fixed subset. The evaluation split is queried repeatedly by the optimizer. Consequently, the reported RMSE measures optimization of this fixed benchmark objective. Assessing generalization to unseen patients would require a separate patient-disjoint evaluation and training-only preprocessing.

Both application domains are mapped from normalized coordinates to $[0,1]^D$ before evaluation. EES and the random pool share the same embedding map and latent bounds. Direct methods search the full original domain. The five application seeds are 101, 202, 303, 404, and 505. Each of the twelve (s, d) combinations is evaluated on these same five seeds.

F.2 Changing the latent bounds and dimension

Figure 4 in the main text shows the selected optimization curves. Figure 12 shows the latent-size grid. The best observed settings occur below the largest tested pair, illustrating the value of matching latent size to the task.

Table 13 gives the final values and curve means for every method displayed in Figure 4. The EES settings show the shared numerical choice $(3, 10)$, Rover's lowest observed final mean at $(2, 30)$, and the low SVR curve mean at $(2, 10)$. These choices use the same five-seed grid, so their apparent advantage includes configuration selection.

For $(3, 10)$, the mean final differences against Vanilla BO on Rover and spherical linear BO on SVR are -1.047 and -0.00369, with three of five paired wins in both cases. The differences in curve mean are -3.244 and -0.0306, with all five pairs favoring EES. With five pairs, even unanimous signs have a minimum two-sided sign-randomization p-value of 0.0625. These descriptive comparisons motivate confirmation on independent seeds.

Early portions also reflect initialization: EES uses five initial points, while Vanilla BO uses 24 on Rover and 60 on SVR. The curves therefore compare complete methods, including their distinct initial designs.

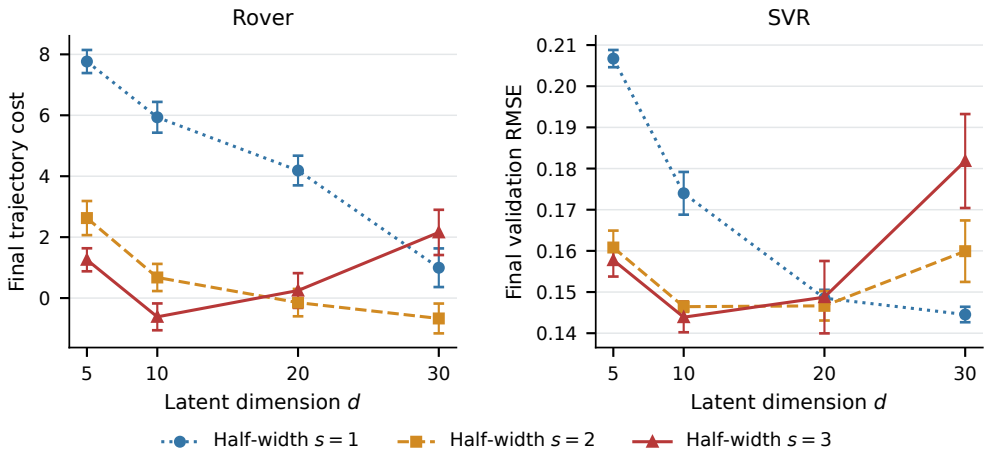

Figure 12: Application grid: all twelve combinations of latent dimension and half-width, using the same five seeds and full search budget per combination (Table 1). Points show mean final subsequent BO values, with one standard error. Differences include changes to the learned matrix and the latent BO problem. Settings used in Figure 4 are selected from this same grid.

Table 13: Application performance over five runs, mean ± standard deviation. Curve mean averages the best-so-far objective over subsequent BO. EES additionally uses the embedding-search budget in Table 1. DSEBO denotes the reproduction variant. All values are minimized. Bold marks the lowest mean for each application and metric.

Application	Method	Final value	Curve mean
Rover	EES $(s, d) = (3, 10)$	-0.617 ± 0.985	$\mathbf{2.082} \pm 1.599$
	EES $(2, 30)$	$\mathbf{-0.669} \pm 1.095$	3.289 ± 0.942
	Vanilla BO	0.430 ± 1.863	5.326 ± 1.815
	Spherical linear BO	2.862 ± 1.251	6.421 ± 1.415
	TuRBO-1	3.384 ± 1.600	6.093 ± 1.646
	BAxUS	4.985 ± 1.954	9.117 ± 0.406
	DSEBO	6.402 ± 0.904	9.895 ± 0.575
	Sobol	9.006 ± 1.133	10.536 ± 0.617
SVR	EES $(s, d) = (3, 10)$	$\mathbf{0.1439} \pm 0.0082$	0.1639 ± 0.0050
	EES $(2, 10)$	0.1464 ± 0.0029	$\mathbf{0.1593} \pm 0.0028$
	Spherical linear BO	0.1476 ± 0.0145	0.1945 ± 0.0103
	Vanilla BO	0.1822 ± 0.0167	0.2179 ± 0.0055
	BAxUS	0.2120 ± 0.0142	0.2237 ± 0.0086
	TuRBO-1	0.2121 ± 0.0071	0.2241 ± 0.0047
	DSEBO	0.2132 ± 0.0097	0.2224 ± 0.0044
	Sobol	0.2290 ± 0.0015	0.2319 ± 0.0030

F.3 Random embeddings and total evaluation cost

The initial application setting, $(s, d) = (1, 5)$, compares EES with an equal-cost random embedding pool and direct Sobol sampling. Table 14 shows that EES improves over the random pool on both objectives. On Rover, direct Sobol performs better at this small latent size, motivating the wider range of settings in Figure 12. The expanded grid examines the joint effect of latent size and the learned matrix.

F.4 What does BO add after embedding search?

Starting subsequent BO with new observations tests the usefulness of the selected embedding. A deployed optimizer could additionally retain the best solution found during embedding search. Table 15 therefore separates the best value from embedding search, the best subsequent BO value, and

Table 14: Initial application setting, $(s, d) = (1, 5)$, over five runs. EES and the random pool report their best subsequent BO values. Sobol reports its best over the same total cost, including embedding search for the embedding methods (Table 1). The last column gives paired EES wins over the random pool. Bold marks the lowest mean.

Application	EES	Random pool	Sobol	EES wins
Rover	7.765	8.046	**3.329**	4/5
SVR	**0.2067**	0.2209	0.2183	5/5

the smaller of the two within each run. At $(s, d) = (3, 10)$, BO improves on the best value from embedding search in all five Rover runs and four of five SVR runs. The combined SVR value is slightly better because embedding search supplies the winner in the remaining run. Rover values are the recorded noisy responses. Independent repeated evaluation would be needed to estimate the expected objective of a returned solution.

Table 15: Best recorded values at $(s, d) = (3, 10)$, averaged over five runs. The combined result takes the minimum within each run before averaging. All columns refer to the same EES runs. Bold marks the lowest mean, including ties.

Application	Embedding search	Subsequent BO	Combined	BO improves
Rover	0.691	$\mathbf{-0.617}$	$\mathbf{-0.617}$	5/5
SVR	0.1477	0.1439	**0.1433**	4/5

Retaining the best earlier solution and supplying earlier observations to the GP are distinct choices. Appendix G examines the latter in an earlier study. Comparing BO with random or evolutionary optimization inside the same selected embedding would further clarify how best to use the subsequent evaluation budget.

F.5 A DIAGNOSTIC OF LOCAL DISTORTION

Proposition 3 explains how to measure the direction most affected by coordinate clipping. To apply this diagnostic, we evaluate each selected embedding on 256 common Sobol points and average the largest eigenvalue of the loss matrix. At the largest setting $(s, d) = (3, 30)$, almost every sampled point has a clipped coordinate on both applications. Nevertheless, the average directional loss is 0.986 on Rover and 0.294 on SVR. Table 16 translates this into the retained squared length of the most affected direction. The difference shows what a directional diagnostic adds to the clipping frequency. Testing its causal relationship with optimization performance would require controlled changes to the embedding geometry.

Table 16: Local geometry at $(s, d) = (3, 30)$, averaged over five selected embeddings. Retained squared length is 1 minus directional loss.

Application	Mean directional loss	Mean retained squared length
Rover	0.986	0.014
SVR	0.294	0.706

F.6 A SMALLER EMBEDDING-SEARCH BUDGET ON NEW SEEDS

The follow-up retains the previously selected $(s, d) = (3, 10)$ configuration, uses ten search generations, and evaluates ten new seeds per application: 606, 707, 808, 909, 1010, 1111, 1212, 1313, 1414, and 1515. EES and the random pool use equal embedding-search and subsequent BO budgets (Table 4). Sobol spends their full combined budget directly. The direct BO reference, with the smaller total budget, is Vanilla BO for Rover and spherical linear BO for SVR.

EES remains better than equal-budget Sobol on every run, while the intervals against the random pool include zero on both tasks. On SVR, EES improves earlier during subsequent BO, with a

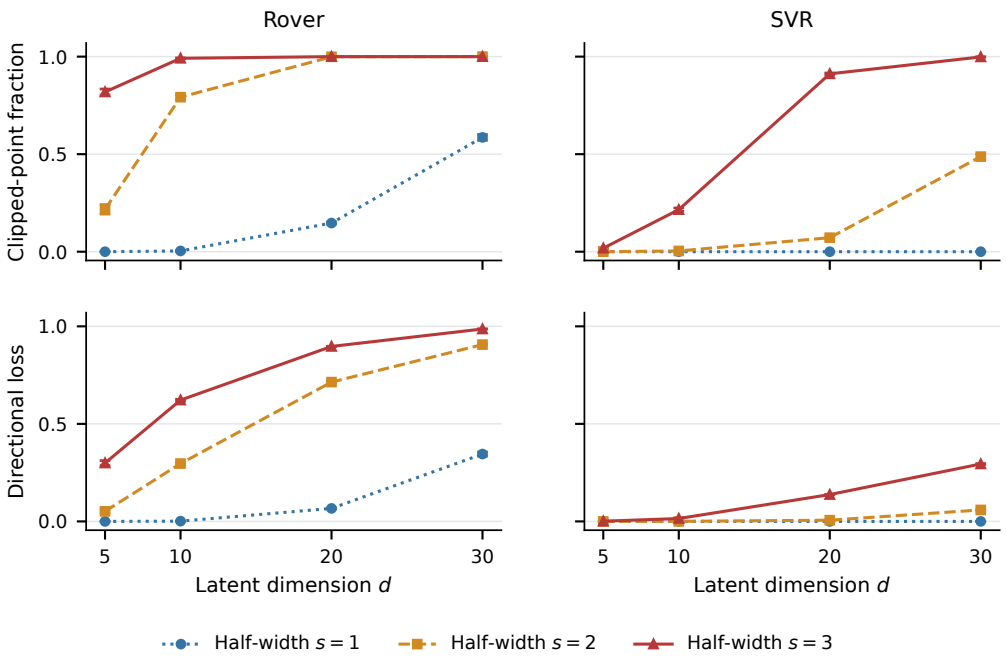

Figure 13: Geometry across the full application grid. The upper panels show the fraction of diagnostic points with a clipped coordinate. The lower panels show the largest local squared-length loss, where zero means no loss and one means a collapsed direction. Points are means over five selected embeddings, with one standard error. Frequent clipping can produce different directional losses on the two applications.

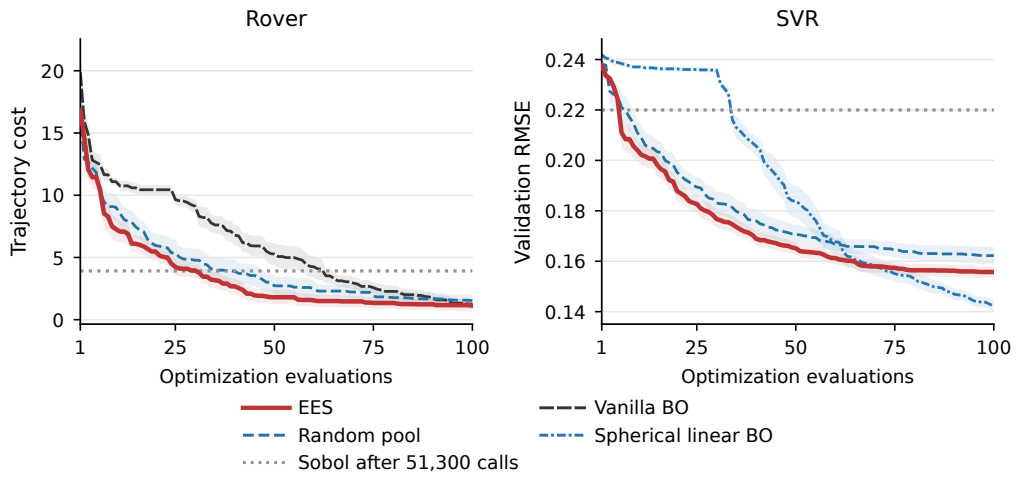

Figure 14: Reduced-budget follow-up on ten new seeds. Lines show mean best-so-far values and bands one standard error. EES and the random pool additionally use ten generations of embedding search (Table 4). The direct BO baseline uses only the displayed evaluations. The horizontal Sobol reference gives its final result after spending the full combined budget directly.

curve-mean difference of -0.0175 against spherical linear BO, but loses on the final value in all ten runs (Figure 14). The distinction between early progress and final value matters when allocating the search budget. Because the seed cohort also changes, a matched-cohort comparison is needed to isolate the effect of reducing embedding search. Figure 15 shows the individual paired differences behind the random-pool comparison.

Table 17: Reduced-budget final differences, EES minus comparator, with paired bootstrap 95% intervals. Negative values favor EES. The direct BO rows have a smaller total budget than EES.

Application	Comparator	Mean difference	95% interval	EES wins
Rover	Random pool	-0.401	$[-1.576, 0.727]$	5/10
	Equal-budget Sobol	-2.784	$[-3.632, -2.012]$	10/10
	Vanilla BO	-0.125	$[-1.285, 1.125]$	6/10
SVR	Random pool	-0.00652	$[-0.01383, 0.00089]$	7/10
	Equal-budget Sobol	-0.06427	$[-0.06890, -0.05925]$	10/10
	Spherical linear BO	0.01348	$[0.01013, 0.01643]$	0/10

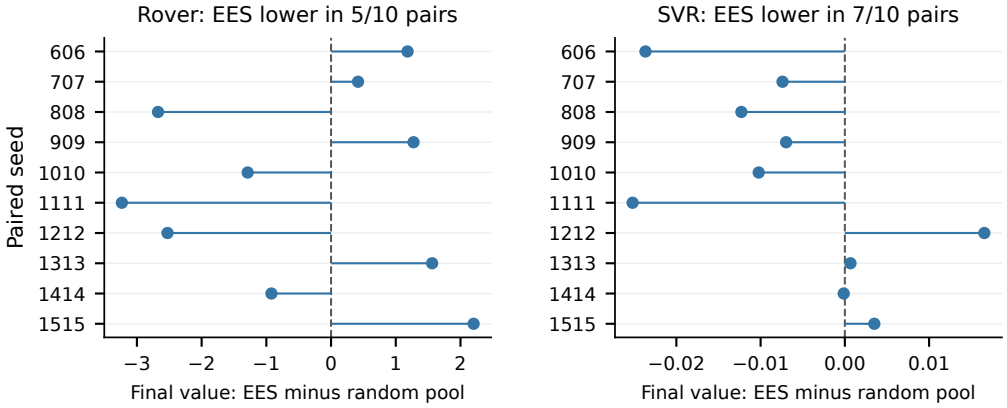

Figure 15: Final differences against the random pool on the new seeds, at equal total budgets (Table 4, $G = 10$). Each point represents one paired run. Values left of zero favor EES. The paired variation shows how outcomes vary around the lower EES mean.

G Observation reuse in an earlier embedding-search study

Starting subsequent BO with new observations allows us to assess the usefulness of the selected embedding separately from retaining a favorable search observation. An earlier study examined whether carrying observations into subsequent BO helps optimization. We report the three cohorts with complete run histories to distinguish two forms of reuse and their different effects.

Historical protocol. This study used an earlier search procedure that scored each embedding with a short BO trajectory. It predates the shared Sobol designs used in the main experiments. Ten candidates were evaluated for ten generations, with 20 evaluations per candidate. The score combined normalized incumbent-curve average, best response, top-five average, and clipping penalty with weights 0.50, 0.30, 0.15, and 0.05. Five shortlisted embeddings were then compared using three validation trajectories of ten evaluations each. The selected embedding received 500 new evaluations in subsequent BO. Thus every run used 2,000 search evaluations, 150 validation evaluations, and 500 subsequent BO evaluations. The 500-evaluation horizon belongs to this separate historical study.

The objectives were 50-dimensional Sphere and Ackley on $[-5, 5]^{50}$ and Rosenbrock on $[0, 2]^{50}$. The latter bounds place the Rosenbrock optimum at the center of the domain. All three objectives have minimum zero and use latent dimension $d = 5$ with bounds $[-1, 1]^5$. In each case, the optimum is accessible at $z = 0$ for every embedding. These centered objectives therefore examine optimization within an accessible region. We used ten seeds per objective and policy, for 90 runs in total. For each seed, the complete search and validation histories and the selected embedding were identical across policies. Subsequent BO used a GP with LogEI. Acquisition optimization used four restarts and 64 raw samples for Sphere, and ten restarts and 128 raw samples for Ackley and centered Rosenbrock.

Reuse policies. *No reuse* starts with five new uniformly sampled latent points, included in the 500-evaluation budget. *Selected-embedding reuse* initializes the GP with observations already obtained through the selected embedding, including its validation observations. *Cross-embedding reuse* additionally admits observations from other embeddings. Writing x_n for an observed point rescaled to $[-1, 1]^D$, the acceptance rule is $\|x_n - B^* B^{*\top} x_n\|_2 / \sqrt{D} \leq 0.1$. These observations are assigned latent coordinates $B^{*\top} x_n$, clipped to the latent bounds where necessary, while retaining the original response. The selected-embedding policy inherits 50–150 observations. The cross-embedding policy inherits 945–1,575 observations. Both reuse policies start acquisition immediately and obtain 500 new evaluations. They also inherit the best response in their training data as the LogEI incumbent. The comparison therefore measures the combined warm-start policy, including its training data, incumbent, and initialization.

Table 18: Historical reuse comparison. Entries give the best objective among the 500 newly evaluated points, averaged over ten seeds, with one standard deviation. Earlier observations are excluded from this metric. Bold marks the lowest mean in each row.

Objective	No reuse	Selected embedding	Selected + others
Sphere	0.00461 ± 0.00491	$\mathbf{0.000389} \pm 0.000256$	0.923 ± 1.079
Ackley	$\mathbf{0.640} \pm 0.293$	3.220 ± 0.227	4.141 ± 0.686
Centered Rosenbrock	0.0627 ± 0.0498	$\mathbf{0.0597} \pm 0.0580$	25.154 ± 11.892

Results and implications. Selected-embedding reuse improves the final Sphere result in all ten paired runs and gives a slightly lower mean on centered Rosenbrock, where it wins seven of ten pairs. These endpoint gains coexist with higher mean best-so-far values averaged across the full run: 0.770 versus 0.569 on Sphere, and 17.289 versus 12.613 on centered Rosenbrock (Figure 16). On Ackley, starting with new observations gives a lower final value in every pair, and the reuse curves plateau at higher objective values. Compared with no reuse, cross-embedding reuse gives worse final new-point results in all thirty paired runs. A small distance between ambient points can still correspond to a substantial difference in objective values, so transferring a response to another embedding requires more than a geometric proximity test. The present comparisons leave this mechanism entangled with training-set size, initialization, and the inherited incumbent.

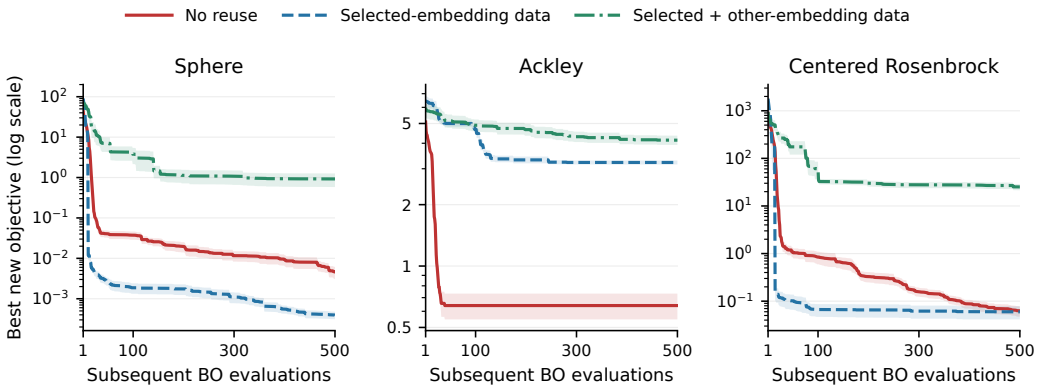

Figure 16: Subsequent BO under the three historical reuse policies. Lines show the mean best objective among newly evaluated points, with one standard error over ten seeds. The vertical axes use a logarithmic scale. The inherited observations influence the reuse policies' GP models and acquisition functions but are excluded from the plotted incumbent. All policies have the same search and validation history within each seed.

Retaining a previously observed solution is a separate decision from training the GP on its response. When all search, validation, and subsequent BO observations are eligible for the final recommendation, the mean best values for no reuse, selected-embedding reuse, and cross-embedding reuse are respectively 0.00383, 0.000389, and 0.0102 on Sphere, 0.303, 0.319, and 0.319 on Ackley, and 0.0598, 0.0531, and 0.271 on centered Rosenbrock. These values show why the deterioration in newly generated points should be distinguished from discarding an already known solution.

The mixed outcomes motivate treating observation reuse as a separate design choice. The main experiments use new observations to evaluate the selected embedding's contribution to subsequent BO. Developing a reuse policy for the current shared-design EES procedure remains a promising way to use the search data more fully.

H Limitations

Embedding search remains costly, so the gains in subsequent BO do not establish an end-to-end efficiency advantage. Longer BO runs, independent assessment of application settings, and broader benchmarks are needed to establish generality. Future work should evaluate cheaper embedding search, adaptive latent size, and selective observation reuse under equal total budgets.