UNCERTAINTY-AWARE OFFLINE DATA-DRIVEN MULTI-OBJECTIVE OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

In offline data-driven Multi-Objective Optimization (MOO), optimization is performed using surrogate models trained only on an offline dataset. When only limited data are available, surrogate models may exhibit substantial epistemic uncertainty and prediction errors. These errors can lead to incorrect dominance judgments and mislead the search process. Existing uncertainty-aware offline data-driven Multi-Objective Evolutionary Algorithms (MOEAs) often rely on Gaussian Process Regression (GPR), limiting their direct applicability to other surrogate models. Meanwhile, generative-based methods may struggle to explore beyond the offline data distribution when it is far from the true Pareto front. We propose EBU-DR, which combines an Empirical-Bayes Uncertainty (EBU) adjustment with a Dual-Ranking (DR) strategy for evolutionary survival selection. EBU uses five-fold out-of-fold predictions to estimate the shrinkage parameters. When predictive signal is sufficient, it shrinks predictions toward the offline objective mean; otherwise, uncertainty estimates are used directly. DR performs non-dominated sorting on both surrogate-predicted and uncertainty-aware objective values, while crowding distances are calculated using only the surrogate-predicted objectives. We evaluate EBU-DR with four surrogate configurations across 29 benchmark problems and different offline dataset sizes, comparing it with representative offline MOO methods.

1 INTRODUCTION

Many real-world Multi-objective Optimization Problems (MOPs) lack analytical functions or simulation models that allow for direct evaluation Mazumdar et al. (2022). Consequently, data-driven Multi-Objective Evolutionary Algorithms (MOEAs) have gained prominence by combining learning-based techniques with evolutionary search. These methods have been applied to trauma systems, airfoil design, energy systems, and smart building optimization Jin et al. (2018); Wang et al. (2018); Liu et al. (2022); Lyu et al. (2025).

In some real-world problems, expensive simulations, time-consuming experiments, or data privacy concerns often result in small datasets. When only limited data are available for surrogate training, epistemic uncertainty in surrogate predictions becomes a critical challenge. The resulting prediction errors can mislead the search toward suboptimal solutions Wang et al. (2018); Smith & Winkler (2006). As shown in Figure 1, relying solely on surrogate-based evaluations f_{sur} can lead to an incorrect dominance judgment.

In the Evolutionary Computation (EC) community, this setting is commonly addressed using offline data-driven MOEAs. To mitigate surrogate predictive uncertainty, one research direction employs ensemble methods such as selective surrogate ensembles Wang et al. (2018), perturbation-based ensemble surrogates Li et al. (2020), and surrogate ensembles based on tri-training Huang et al. (2021). Other methods incorporate uncertainty estimates into the optimization process, including adaptive model selection IBEA-MS Liu et al. (2022), probabilistic selection in Prob-RVEA and Prob-MOEA/D Mazumdar et al. (2022). TGPR-MO Mazumdar et al. (2023) combines regression trees with Gaussian process regression (GPR), while DDMOEA-GAN Zhang et al. (2024a) employs GAN-based data augmentation and trust-aware fitness evaluation during optimization.

In the Machine Learning (ML) community, a related research direction is known as offline Model-Based Optimization (MBO) Trabucco et al. (2021). The offline MOO benchmark Xue et al. (2024)

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.

includes deep-learning methods such as End-to-End, Multiple Models and Multiple Models with Conservative Objective Models (COMs) Trabucco et al. (2021). Recent generative-based methods include ParetoFlow, which uses predictor-guided flow matching Yuan et al. (2025), and Pareto-Conditioned Diffusion (PCD), which directly generates candidates conditioned on target objectives without requiring explicit surrogate models Shrestha et al. (2026). In addition, DOMOO employs out-of-distribution risk control and nested Pareto set learning to obtain diverse, high-quality solutions Zhu et al. (2026).

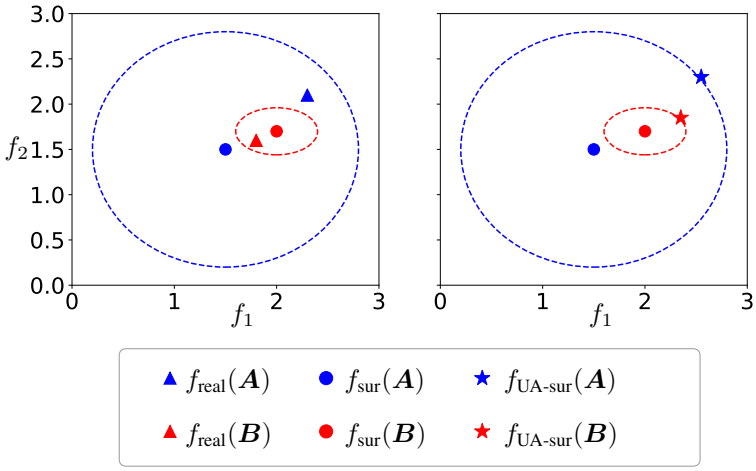


Figure 1: An example of misleading dominance judgments caused by surrogate prediction errors and their mitigation through EBU-DR. The dashed circle denotes the predictive uncertainty interval centered at the surrogate-predicted objective values. Left: Surrogate predictions incorrectly indicate $\boldsymbol{A} \prec \boldsymbol{B}$, whereas the real objective values satisfy $\boldsymbol{B} \prec \boldsymbol{A}$. Right: The uncertainty-aware objective values yield $\boldsymbol{B} \prec \boldsymbol{A}$. By considering both sets of objective values, EBU-DR makes the two solutions non-dominated in the hybrid objective space, preventing the incorrect surrogate-only dominance judgment.

Among the above methods, ensemble-based methods focus only on the surrogate side. Existing uncertainty-aware offline data-driven MOEAs are often designed using GPR, limiting their direct applicability to other surrogates. Vanilla deep learning-based methods and generative-based methods may not perform well on small offline datasets. Moreover, recent evidence shows that generative methods tend to remain close to the offline data distribution and may struggle to explore out-of-distribution regions required to approach the true Pareto front, particularly when the offline dataset is substantially shifted from the Pareto frontier Holly et al. (2026). In contrast, MOEAs can explore beyond the offline data distribution, but the ranking and selection process may be misled by surrogate prediction errors.

To address this challenge, we propose an uncertainty-aware EBU-DR framework for offline data-driven MOO. Our contributions are summarized as follows:

- **Empirical-Bayes Uncertainty Adjustment.** We develop an Empirical-Bayes Uncertainty (EBU) adjustment using out-of-fold predictions to adaptively incorporate predictive uncertainty without additional real evaluations.
- **Dual-Ranking Strategy.** We propose a Dual-Ranking (DR) strategy that considers both surrogate-predicted and uncertainty-aware objective values. The strategy can be flexibly integrated with different surrogate models.
- **Extensive Evaluation.** We evaluate the proposed framework with surrogate models across benchmark problems under different settings and compare these methods with the aforementioned baselines.

2 PRELIMINARIES

2.1 MULTI-OBJECTIVE OPTIMIZATION

We consider the following Multi-objective Optimization Problem (MOP):

$$\begin{aligned} &\text{minimize } \boldsymbol{f}(\boldsymbol{x}) = (f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \ldots, f_K(\boldsymbol{x})) \\ &\text{subject to } \boldsymbol{x} \in S, \end{aligned} \tag{1}$$

where $\boldsymbol{x}$ is the decision vector, $K \geq 2$ is the total number of objectives, and S is the feasible region in the decision space. A solution $\boldsymbol{x}_1 \in S$ dominates another solution $\boldsymbol{x}_2 \in S$, denoted as $\boldsymbol{x}_1 \prec \boldsymbol{x}_2$, if $f_k(\boldsymbol{x}_1) \leq f_k(\boldsymbol{x}_2)$ for all $k = 1, \ldots, K$ and $f_k(\boldsymbol{x}_1) < f_k(\boldsymbol{x}_2)$ for at least one $k = 1, \ldots, K$. The set of solutions that cannot be dominated by any other feasible solution is the Pareto set, and the set of their objective values is the Pareto front.

2.2 OFFLINE DATA-DRIVEN VS. ONLINE SURROGATE-ASSISTED MULTI-OBJECTIVE OPTIMIZATION

In offline data-driven multi-objective optimization (MOO), surrogate models are trained on a fixed, pre-collected dataset and used to guide the search without further real objective evaluations Liu et al. (2022); Mazumdar et al. (2022). In contrast, online surrogate-assisted MOO can evaluate selected candidate solutions using the real objective functions and update the surrogate models with newly acquired data Zhang et al. (2024b).

2.3 SURROGATE MODELING

Each objective is approximated by a surrogate model $\mathcal{M}$ trained on the fixed offline dataset $\mathcal{D}$. We consider four surrogate configurations: GPR with ARD RBF Williams & Rasmussen (2006) and Matérn 5/2 kernels Snoek et al. (2012), Quantile Regression (QR) Koenker & Hallock (2001), and Bayesian Neural Networks (BNN) Blundell et al. (2015). GPR provides a predictive mean and latent-function standard deviation, and is widely used in surrogate-based optimization, but suffers from high computational cost on large datasets. QR provides the median and upper conditional quantiles without assuming a Gaussian predictive distribution. BNN provides posterior means, standard deviations, and upper quantiles of the latent function, enabling flexible nonlinear modeling but incurring high training and inference costs. These predictions and uncertainty estimates are used in the uncertainty-aware ranking methods. Detailed surrogate formulations are provided in Appendix.

3 METHOD

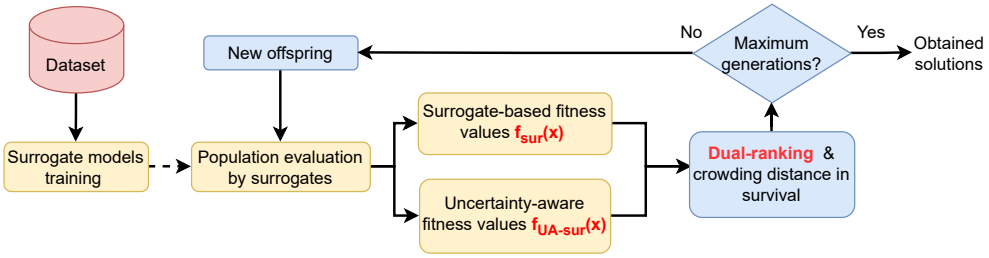


Figure 2: Workflow of surrogate training and optimization. EBU-DR is implemented by performing non-dominated sorting on solutions using both the surrogate-based function f_{sur} and the uncertainty-aware function $f_{\text{UA-sur}}$.

3.1 MOTIVATION

We assume that data points $x_1, x_2, \ldots, x_n$ in a dataset $\mathcal{D}$ are drawn from an unknown distribution $p(x)$. During surrogate training, the models aim to reduce prediction errors over the dataset. However,

during optimization, an off-the-shelf multi-objective optimization algorithm (e.g., NSGA-II Deb et al. (2002)) explores the input space $\boldsymbol{X}$ to minimize the objective $\boldsymbol{f}(\boldsymbol{x})$, relying solely on the surrogate-based evaluation. When the optimizer explores unseen areas, *i.e.* candidate solutions located in low-density regions of the distribution $p(x)$, surrogate predictions may become unreliable and overly optimistic. Such solutions may be falsely ranked as non-dominated, as illustrated in Figure 1. To address this issue, we consider both predicted values and uncertainty estimates from surrogates. The detailed workflow is shown in Figure 2.

3.2 Uncertainty-Aware Function

For the k-th objective of a candidate solution $\boldsymbol{x}$, the surrogate model $\mathcal{M}_k$ outputs a predicted objective value and an uncertainty estimate. Their specific forms depend on the surrogate model. We define the surrogate-predicted objective function as:

$$f_{\text{sur},k}(\boldsymbol{x}) = \begin{cases} \mu_k(\boldsymbol{x}) & \text{(GPR)}, \\ q_{0.5,k}(\boldsymbol{x}) & \text{(QR)}, \\ \mu_k^{\text{BNN}}(\boldsymbol{x}) & \text{(BNN)}, \end{cases} \tag{2}$$

where μ_k is the GPR predictive mean, $q_{0.5,k}$ is the QR median prediction, and μ_k^{BNN} is the BNN posterior sample mean.

We use $s_k(\boldsymbol{x})$ to represent uncertainty estimates from different surrogate models. GPR provides the latent-function predictive standard deviation. BNN provides the standard deviation of Monte Carlo latent-function samples drawn from its variational posterior. For QR, a non-negative standard-deviation proxy is constructed from the median and upper quantile: $s_k(\boldsymbol{x}) = \frac{\max(q_{0.9,k}(\boldsymbol{x}) - q_{0.5,k}(\boldsymbol{x}),\, 0)}{\Phi^{-1}(0.9)}$, where Φ^{-1} denotes the inverse standard normal Cumulative Distribution Function (CDF). This conversion provides a common uncertainty representation.

3.2.1 Empirical-Bayes Uncertainty Adjustment

Building on empirical-Bayes shrinkage Efron & Morris (1973), we develop an Empirical-Bayes Uncertainty (EBU) adjustment to utilize both uncertainty estimates and the offline dataset. For each objective, we perform five-fold cross-validation on the offline dataset to obtain Out-Of-Fold (OOF) predictions and uncertainty estimates. These OOF predictions are then used to estimate specific adjustment parameters.

For the k-th objective, let m_k denote the mean of real objective values in offline dataset, τ_k^2 the estimated signal variance, and c_k an uncertainty scaling parameter. We define the adjustment weight as:

$$w_k(\boldsymbol{x}) = \frac{\tau_k^2}{\tau_k^2 + c_k \min\left(s_k^2(\boldsymbol{x}), s_{\max,k}^2\right)}, \tag{3}$$

where $s_{\max,k}^2$ is the maximum squared OOF uncertainty estimate for the k-th objective. The clipping operation $\min(\cdot)$ limits the influence of unusually large uncertainty estimates $s_k^2(\boldsymbol{x})$ on the adjustment weight.

The resulting uncertainty-aware function is defined as:

$$f_{\text{UA-sur},k}(\boldsymbol{x}) = \begin{cases} m_k + w_k(\boldsymbol{x})\left(f_{\text{sur},k}(\boldsymbol{x}) - m_k\right), & \text{if } \widehat{\text{Cov}}_k > \text{SE}(\widehat{\text{Cov}}_k), \\ s_k(\boldsymbol{x}), & \text{otherwise}, \end{cases} \tag{4}$$

where $\widehat{\text{Cov}}_k$ is the estimated covariance between fold-centered OOF predictions and real objective values, and $\text{SE}(\widehat{\text{Cov}}_k)$ is an empirical variability estimate. When sufficient predictive signal is detected, the EBU adjustment shrinks surrogate predictions toward the mean of real objective values in offline dataset. Otherwise, the uncertainty estimate itself is used. Detailed parameter estimation procedures are provided in the Appendix.

3.3 Dual-Ranking Strategy

During survival selection, the surrogate-predicted objective values $f_{\text{sur}}(\boldsymbol{x})$ and the uncertainty-aware objective values $f_{\text{UA-sur}}(\boldsymbol{x})$ are concatenated to form a hybrid objective vector:

$$f_{\text{hyb}}(\boldsymbol{x}) = [f_{\text{sur}}(\boldsymbol{x}), f_{\text{UA-sur}}(\boldsymbol{x})] \in \mathbb{R}^{2K}. \tag{5}$$

Non-dominated sorting is performed in the $2K$-dimensional hybrid objective space, while crowding distances are calculated using only the original K surrogate-predicted objectives. Survivors are selected according to their non-dominated ranks and crowding distances. By incorporating the uncertainty-aware objective values, the strategy allows uncertainty information to modify dominance relationships established by surrogate predictions alone. It can also be flexibly integrated with different surrogate models. The full procedure is provided in Algorithm 1 in Appendix A.3.

3.4 Time Complexity

We analyze the time complexity of the proposed DR strategy when used with NSGA-II. The standard non-dominated sorting procedure has a time complexity of $O(KN^2)$, where N denotes the number of candidate solutions and K the number of objectives. In the proposed DR strategy, the time complexity of concatenation operator is $O(KN)$. Non-dominated sorting is performed in the $2K$-dimensional objective space, resulting in a worst-case time complexity of $O(2KN^2)$. Therefore, total time complexity is $O(KN) + O(2KN^2) \approx O(KN^2)$.

EBU-DR additionally requires five-fold cross-validation to estimate its adjustment parameters. This preprocessing cost depends on the surrogate model and is incurred once per offline dataset and surrogate configuration.

4 Experiments

4.1 Experimental Setup

4.1.1 Benchmark Problems

We select 29 benchmark problems from the Offline MOO Benchmark Xue et al. (2024) that fit the multi-objective optimization setting (2 and 3 objectives) and are suitable for the small-data setting. The selected problems include ZDT1, ZDT2, ZDT3, ZDT4, ZDT6, OmniTest, VLMOP1–VLMOP3, DTLZ1–DTLZ7, RE21–RE25, RE31–RE37, and MO-Portfolio.

4.1.2 Offline Dataset

We use the official training pools provided by the Offline MOO Benchmark Xue et al. (2024). For each benchmark problem, 10 random permutations of the training data are generated using seeds 1–10. For each permutation, the first N samples are selected to construct an offline dataset, where $N \in \{50, 100, 200, 400, 1000\}$. Thus, 50 offline datasets with different sample sizes and empirical distributions are generated for each problem to comprehensively evaluate the methods. The test dataset is kept separate from model training and is used only for evaluating surrogate prediction performance.

4.1.3 Baseline Methods

Four offline data-driven methods are Prob-RVEA and Prob-MOEA/D Mazumdar et al. (2022), TGPR-MO Mazumdar et al. (2023), and DDMOEA-GAN Zhang et al. (2024a). Three classic deep-learning methods and one Bayesian optimization method selected from the Offline MOO Benchmark Xue et al. (2024) are End2End-Vanilla, MultipleModels-Vanilla, MultipleModels-COM, and MOBO-Vanilla. We additionally include two recent generative-based methods, PCD Shrestha et al. (2026) and ParetoFlow Yuan et al. (2025).

All methods are trained on the same offline datasets; method-specific training and search mechanisms follow their original implementations or the adaptations detailed in the Appendix.

4.1.4 Surrogate Models and Optimization Algorithm

Our proposed method is combined with four different surrogate models, including GPR with RBF and Matérn 5/2 kernels, quantile regression (QR), and Bayesian neural networks (BNN), together with the well-known MOEA NSGA-II Deb et al. (2002). NSGA-II is implemented using pymoo Blank & Deb (2020). The population size is set to 100, and the maximum number of generations is 100, resulting in a total of 10,000 fitness evaluations. Optimization seeds are set to 1–10. Detailed surrogate and optimization algorithm configurations are provided in the Appendix.

4.1.5 Performance Indicators

Optimization performance is evaluated using Mean Squared Error (MSE), Inverted Generational Distance Plus (IGD+) Ishibuchi et al. (2015), and Hypervolume (HV) Zitzler & Thiele (1999). MSE measures the discrepancy between surrogate-predicted and real objective values of the final solutions. IGD+ and HV evaluate the quality of the final solutions based on their real objective values. Lower is better for MSE and IGD+, and higher is better for HV. Normalization and HV reference points are provided in the Appendix.

4.2 Ablation Study

The ablation study evaluates the overall effect of EBU-DR and the additional contribution of the proposed EBU adjustment. For each of the four surrogate configurations, we compare three variants: standard NSGA-II using only surrogate-predicted objectives (Normal), NSGA-II with DR, and NSGA-II with EBU-DR. For DR, the second objective view uses the 0.9 upper predictive quantile. Specifically, GPR uses $\mu_k(\boldsymbol{x}) + \Phi^{-1}(0.9)s_k(\boldsymbol{x})$, while QR and BNN use their respective 0.9 quantile predictions.

All three variants use the same trained surrogate, offline dataset, and optimization settings. Each surrogate configuration is evaluated on 145 combinations of benchmark problems and offline dataset sizes. Results are averaged over 100 paired combinations of offline data and optimization seeds for each benchmark problem–dataset size combination. We compare EBU-DR with Normal to evaluate the overall effect of the proposed framework, and with DR to assess the additional contribution of the EBU adjustment.

Table 1 shows the ablation results. Compared with Normal, EBU-DR achieves more wins than losses for every indicator in the aggregate: $258/113$ for MSE, $217/131$ for IGD+, and $228/143$ for HV. All four surrogate configurations have more wins than losses for each indicator, although QR is nearly balanced on HV and QR and BNN exhibit many numerical ties. Thus, EBU-DR achieves better mean performance in more benchmark settings than standard NSGA-II using only surrogate-predicted objectives.

When EBU-DR compares with DR, EBU-DR achieves better mean IGD+ and HV in more benchmark settings than DR. This advantage is concentrated in the GPR surrogate models. DR achieves lower MSE more frequently across all four surrogate configurations. The results suggest that better optimization performance does not necessarily correspond to lower surrogate prediction error at the obtained solutions, particularly for GPR surrogates.

Many ties with Normal are consistent with the EBU adjustment reducing to the original predictions. For 68 QR and 78 BNN problem-dataset size combinations, all 10 offline-data seeds have sufficient predictive signal and yield $c_k = 0$ for each objective. Consequently, $w_k(\boldsymbol{x}) = 1$ and $f_{\text{UA-sur},k} = f_{\text{sur},k}$, so concatenating the two objective functions leaves dominance relationships unchanged. This explains many of the numerical ties and why the EBU adjustment may leave the search unchanged for some offline datasets.

4.3 Main Results

We compare the four EBU-DR surrogate configurations with the offline MOO baselines across all benchmark settings. For each benchmark setting, we average the results over 100 runs. The selected offline subsets, objective normalization bounds, HV reference points, and IGD+ reference sources match across methods.

Table 1: Ablation results for EBU-DR across 29 benchmark problems and 5 different offline dataset sizes. The entries $+/=/-$ indicate the numbers of benchmark problem–dataset size combinations in which EBU-DR achieves better, numerically tied, or worse mean values than the comparator (Normal or DR). A numerical tie satisfies $|a-b| \leq 10^{-12} + 10^{-9}|b|$, where a and b are the mean values of EBU-DR and the comparator, respectively. Bold indicates a win count greater than both the tie and loss counts for the same indicator.

Comparison	Surrogates	MSE			IGD+			HV		
		$+$	$=$	$-$	$+$	$=$	$-$	$+$	$=$	$-$
EBU-DR vs. Normal	GPR (RBF)	**82**	32	31	**57**	41	47	**60**	32	53
	GPR (Matérn)	**86**	25	34	**70**	35	40	**76**	25	44
	QR	46	70	29	42	72	31	38	70	37
	BNN	44	82	19	48	84	13	54	82	9
	Overall	**258**	209	113	217	232	131	**228**	209	143
EBU-DR vs. DR	GPR (RBF)	52	1	92	**84**	12	49	**96**	0	49
	GPR (Matérn)	36	0	109	**95**	13	37	**103**	0	42
	QR	60	0	85	71	0	74	69	0	76
	BNN	49	0	96	63	3	79	70	0	75
	Overall	197	1	382	**313**	28	239	**338**	0	242

For each benchmark setting, methods are ranked separately according to each performance indicator. For methods with all three indicators available, the overall mean rank is calculated as the equally weighted average of the three indicator-specific ranks without rescaling. Table 2 reports the mean and standard deviation of these ranks across all benchmark settings. This avoids pooling raw indicator values from differently scaled problems. Numerical ties use the tolerance in Table 1 and receive average ranks. PCD does not produce surrogate objective predictions, so its MSE is not applicable. MOBO is temporarily excluded because the available results are insufficient for benchmark comparisons.

Table 2: Indicator-specific ranks (mean $\pm$ standard deviation) across all benchmark settings. Smaller ranks are better. Bold and underline indicate the best and second-best mean ranks in each column. PCD's unavailable MSE and overall mean rank are shown as dashes. Rank standard deviations reflect variation across benchmark problems and dataset sizes.

Methods	MSE rank	IGD+ rank	HV rank	Overall mean rank
Prob-RVEA	5.73 ± 3.11	5.66 ± 3.14	5.12 ± 3.21	5.51
Prob-MOEA/D	$\mathbf{3.83 \pm 3.04}$	8.50 ± 3.32	8.71 ± 3.41	7.01
TGPR-MO	6.64 ± 2.64	6.45 ± 3.12	6.56 ± 3.09	6.55
DDMOEA-GAN	8.48 ± 3.99	10.54 ± 3.46	11.33 ± 2.70	10.12
End-to-End	7.49 ± 2.42	$\underline{4.87 \pm 3.24}$	5.13 ± 3.09	5.83
Multiple Models	7.81 ± 2.59	$\mathbf{4.44 \pm 3.41}$	4.82 ± 3.33	5.69
Multiple Models with COMs	6.79 ± 3.75	8.67 ± 3.25	9.41 ± 2.71	8.29
ParetoFlow	$\underline{4.98 \pm 2.85}$	8.42 ± 2.82	9.17 ± 2.40	7.52
PCD	—	6.94 ± 3.94	7.34 ± 3.76	—
GPR (RBF)+EBU-DR	5.32 ± 3.50	5.75 ± 3.26	$\underline{4.73 \pm 3.10}$	$\underline{5.27}$
GPR (Matérn)+EBU-DR	5.01 ± 2.80	5.02 ± 2.87	$\mathbf{3.96 \pm 2.52}$	**4.66**
QR+EBU-DR	7.60 ± 3.41	6.64 ± 3.25	6.02 ± 2.65	6.75
BNN+EBU-DR	8.33 ± 3.26	9.10 ± 3.17	8.69 ± 3.26	8.71

Among the 12 methods with all three indicators available, GPR (Matérn)+EBU-DR achieves the lowest overall mean rank (4.66), followed by GPR (RBF)+EBU-DR (5.27). The two configurations also achieve the best and second-best mean HV ranks of 3.96 and 4.73, respectively. However, no method leads all indicators. Multiple Models achieves the best IGD+ rank (4.44), followed by End-to-End (4.87), while GPR (Matérn)+EBU-DR ranks third (5.02). Prob-MOEA/D has the best

MSE rank (3.83), despite its relatively weak HV and IGD+ ranks. These results also suggest that better optimization performance does not necessarily correspond to lower surrogate prediction error at the obtained solutions. ParetoFlow has a relatively favorable MSE rank (4.98) but a weaker HV rank (9.17), further illustrating the distinction between surrogate agreement and optimization quality.

Against Multiple Models, GPR (Matérn)+EBU-DR records $84/0/61$ wins/ties/losses in HV and $116/0/29$ in MSE, but $61/6/78$ in IGD+. Its HV advantage is concentrated in synthetic problems ($58/0/22$) and MO-Portfolio ($5/0/0$); Multiple Models achieves better HV in more RE engineering problems ($39/0/21$). The results therefore support competitive overall HV performance rather than uniform superiority across indicators or problem families.

Overall, the strongest EBU-DR results are obtained with the GPR surrogates, particularly the Matérn kernel.

4.4 Effect of Offline Dataset Size

To examine how data availability affects EBU-DR, we split the results by offline dataset size. At each N, we compare the 100-run means of EBU-DR with Normal and DR on 29 problems for each of the four surrogates.

Table 3 shows that, against Normal, HV wins decrease from 57 at $N = 50$ to 34 at $N = 1000$, while ties increase from 27 to 52. MSE and IGD+ also show fewer wins and more ties at the largest dataset size. In contrast, EBU-DR records more HV and IGD+ wins than losses against DR at every size, but more MSE losses than wins. The HV counts against DR change from $63/0/53$ at $N = 50$ to $71/0/45$ at $N = 1000$, indicating that this advantage is not limited to the smallest datasets.

Table 3: Dataset-size-specific results, aggregated across the four surrogates. Each entry reports EBU-DR's wins/ties/losses against the comparator over the benchmark settings. Comparisons use the 100-run means and same numerical tie criterion.

Comparison	N	MSE (W/T/L)	IGD+ (W/T/L)	HV (W/T/L)
EBU-DR vs. Normal	50	$66/26/24$	$58/29/29$	$57/27/32$
	100	$59/34/23$	$52/37/27$	$56/34/26$
	200	$55/46/15$	$43/50/23$	$45/45/26$
	400	$40/51/25$	$37/58/21$	$36/51/29$
	1000	$38/52/26$	$27/58/31$	$34/52/30$
EBU-DR vs. DR	50	$42/0/74$	$58/4/54$	$63/0/53$
	100	$39/1/76$	$59/4/53$	$66/0/50$
	200	$38/0/78$	$64/6/46$	$65/0/51$
	400	$39/0/77$	$70/7/39$	$73/0/43$
	1000	$39/0/77$	$62/7/47$	$71/0/45$

Overall, EBU-DR's wins over Normal become less frequent as the offline dataset size increases. Compared with DR, EBU-DR achieves more IGD+ and HV wins than losses at each dataset size.

5 Conclusion

In this work, we propose EBU-DR for offline data-driven MOO with limited training data. The EBU adjustment estimates objective-wise parameters from five-fold out-of-fold predictions to construct an uncertainty-aware objective function. The DR strategy then performs non-dominated sorting on both surrogate-predicted and uncertainty-aware objective values, while crowding distances are computed using only surrogate predictions. This framework incorporates uncertainty into survival selection without requiring additional real objective evaluations.

We evaluate EBU-DR with four surrogate configurations across 29 benchmark problems and 5 offline dataset sizes. In the ablation study, EBU-DR achieves more wins than losses against surrogate-only NSGA-II for all three indicators. Compared with fixed-quantile DR, it achieves more wins than losses for IGD+ and HV overall, with the improvements concentrated in the GPR configurations,

although DR more frequently achieves lower prediction error at the obtained solutions. In the baseline comparison, GPR (Matérn)+EBU-DR achieves the lowest mean HV rank (3.96) among 13 methods and the lowest descriptive overall mean rank (4.66) among methods with all three indicators available. These results highlight the distinction between optimization performance and surrogate prediction error at the obtained solutions.

Future work will extend EBU-DR to many-objective optimization ($K > 3$) and evaluate its performance on additional real-world problems. We will also investigate extensions to online settings with continuously arriving data and noisy objective measurements.

Reproducibility statement

The experimental settings, including benchmarks, surrogates, optimization, and evaluation, as well as implementation details, are described in the main text and appendices. Source code and configuration files are provided in the supplementary material.

AI use statement

Generative AI tools were used to assist with methodological exploration, mathematical derivations, code development and debugging, experimental analysis, and manuscript writing and editing. The authors critically reviewed AI-assisted outputs and take full responsibility for the methodology, implementation, experimental results, and final manuscript.

Ethics statement

We do not identify ethical concerns associated with this research.

References

J. Blank and K. Deb. pymoo: Multi-objective optimization in python. *IEEE Access*, 8:89497–89509, 2020.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural network. In *International conference on machine learning*, pp. 1613–1622. PMLR, 2015.

Kalyanmoy Deb, Ram Bhushan Agrawal, et al. Simulated binary crossover for continuous search space. *Complex Systems*, 9(2):115–148, 1995.

Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6(2):182–197, 2002.

Bradley Efron and Carl Morris. Stein's estimation rule and its competitors—an empirical bayes approach. *Journal of the American Statistical Association*, 68(341):117–130, 1973.

Stephanie Holly, Alexandru-Ciprian Zăvoianu, Siegfried Silber, Sepp Hochreiter, and Werner Zellinger. The offline-frontier shift: Diagnosing distributional limits in generative multi-objective optimization. *arXiv preprint arXiv:2602.11126*, 2026.

Pengfei Huang, Handing Wang, and Yaochu Jin. Offline data-driven evolutionary optimization based on tri-training. *Swarm and evolutionary computation*, 60:100800, 2021.

Hisao Ishibuchi, Hiroyuki Masuda, Yuki Tanigaki, and Yusuke Nojima. Modified distance calculation in generational distance and inverted generational distance. In *Evolutionary Multi-Criterion Optimization: 8th International Conference, EMO 2015, Guimarães, Portugal, March 29–April 1, 2015. Proceedings, Part II 8*, pp. 110–125. Springer, 2015.

Yaochu Jin, Handing Wang, Tinkle Chugh, Dan Guo, and Kaisa Miettinen. Data-driven evolutionary optimization: An overview and case studies. *IEEE Transactions on Evolutionary Computation*, 23 (3):442–458, 2018.

Roger Koenker and Kevin F Hallock. Quantile regression. *Journal of economic perspectives*, 15(4): 143–156, 2001.

Jian-Yu Li, Zhi-Hui Zhan, Hua Wang, and Jun Zhang. Data-driven evolutionary algorithm with perturbation-based ensemble surrogates. *IEEE transactions on cybernetics*, 51(8):3925–3937, 2020.

Zhening Liu, Handing Wang, and Yaochu Jin. Performance indicator-based adaptive model selection for offline data-driven multiobjective evolutionary optimization. *IEEE transactions on cybernetics*, 53(10):6263–6276, 2022.

Huanbo Lyu, Daniel Herring, Hongkuan Chen, Shiqiao Zhou, Jianyu Zhang, Lingfeng Wang, Zheming Zuo, James Andrews, Fabian Spill, Jelena Ninic, et al. A data-driven multi-objective optimisation framework for energy efficiency and thermal comfort in flexible building spaces. *Energy and Buildings*, pp. 116100, 2025.

Atanu Mazumdar, Tinkle Chugh, Jussi Hakanen, and Kaisa Miettinen. Probabilistic selection approaches in decomposition-based evolutionary algorithms for offline data-driven multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 26(5):1182–1191, 2022.

Atanu Mazumdar, Manuel López-Ibáñez, Tinkle Chugh, Jussi Hakanen, and Kaisa Miettinen. Treed gaussian process regression for solving offline data-driven continuous multiobjective optimization problems. *Evolutionary computation*, 31(4):375–399, 2023.

Matthias Seeger. Gaussian processes for machine learning. *International journal of neural systems*, 14(02):69–106, 2004.

Jatan Shrestha, Santeri Heiskanen, Kari Hepola, Severi Rissanen, Pekka Jääskeläinen, and Joni Pajarinen. Pareto-conditioned diffusion models for offline multi-objective optimization. In *International Conference on Learning Representations*, 2026.

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

Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. *Advances in neural information processing systems*, 25, 2012.

Ingo Steinwart and Andreas Christmann. Estimating conditional quantiles with the help of the pinball loss. *Bernoulli*, 17:211–225, 2011. URL `https://api.semanticscholar.org/CorpusID:21704877`.

Brandon Trabucco, Aviral Kumar, Xinyang Geng, and Sergey Levine. Conservative objective models for effective offline model-based optimization. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 10358–10368. PMLR, 2021.

Handing Wang, Yaochu Jin, Chaoli Sun, and John Doherty. Offline data-driven evolutionary optimization using selective surrogate ensembles. *IEEE Transactions on Evolutionary Computation*, 23(2):203–216, 2018.

Christopher KI Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, 2006.

Ke Xue, Rong-Xi Tan, Xiaobin Huang, and Chao Qian. Offline multi-objective optimization. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 55595–55624. PMLR, 2024.

Ye Yuan, Can Chen, Christopher Pal, and Xue Liu. Paretoflow: Guided flows in multi-objective optimization. In *International Conference on Learning Representations*, 2025.

Yu Zhang, Wang Hu, Wen Yao, Lixian Lian, and Gary G. Yen. Offline data-driven multiobjective optimization evolutionary algorithm based on generative adversarial network. *IEEE Transactions on Evolutionary Computation*, 28(2):293–306, 2024a. doi: 10.1109/TEVC.2022.3231493.

Zhiyao Zhang, Yong Wang, Jiao Liu, Guangyong Sun, and Ke Tang. A two-phase kriging-assisted evolutionary algorithm for expensive constrained multiobjective optimization problems. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 54(8):4579–4591, 2024b.

Yiyi Zhu, Yaolin Wen, Xiang Xia, Xin An, Hanyi Si, Xiang Shu, Yangde Fu, Liang Dou, and Hong Qian. Diversity-driven offline multi-objective optimization via nested pareto set learning. *arXiv preprint arXiv:2606.15115*, 2026.

Eckart Zitzler and Lothar Thiele. Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. *IEEE transactions on Evolutionary Computation*, 3(4):257–271, 1999.

A Appendix

A.1 Surrogate Model Formulations

The following formulations summarize the surrogate models used in the experiments. Their implementation and training hyperparameters are specified in the Experimental Settings below.

A.1.1 Gaussian Process Regression

GPR models the latent objective function as $f(\boldsymbol{x}) \sim \mathcal{GP}(\mu(\boldsymbol{x}), k(\boldsymbol{x}, \boldsymbol{x}'))$. The two kernels considered in this work are the ARD RBF kernel Seeger (2004) and the ARD Matérn 5/2 kernel Snoek et al. (2012):

$$r^2(\boldsymbol{x}, \boldsymbol{x}') = \sum_{d=1}^{D} \left(\frac{x_d - x'_d}{\ell_d} \right)^2 ,$$

where ℓ_d is the length scale of variable d, their covariance functions are:

$$k_{\mathrm{RBF}}(\boldsymbol{x}, \boldsymbol{x}') = \sigma_f^2 \exp\!\left(-\tfrac{1}{2} r^2(\boldsymbol{x}, \boldsymbol{x}')\right)$$

and

$$k_{\mathrm{M52}}(\boldsymbol{x}, \boldsymbol{x}') = \sigma_f^2 \left(1 + \sqrt{5r^2(\boldsymbol{x}, \boldsymbol{x}')} + \tfrac{5}{3} r^2(\boldsymbol{x}, \boldsymbol{x}')\right) \exp\!\left(-\sqrt{5r^2(\boldsymbol{x}, \boldsymbol{x}')}\right).$$

Observation likelihood noise is excluded when computing the uncertainty used by our methods.

A.1.2 Quantile Regression

QR estimates conditional quantiles without requiring a Gaussian predictive distribution. For a quantile level $\tau \in (0, 1)$, its general pinball-loss is:

$$q_\tau = \arg\min_{g \in \mathcal{G}} \sum_{i=1}^{N} \rho_\tau(y_i - g(\boldsymbol{x}_i)) , \qquad \rho_\tau(u) = u \left(\tau - \mathbb{I}[u < 0]\right),$$

where $\mathcal{G}$ denotes a model class and ρ_τ is the check loss Steinwart & Christmann (2011). This expression describes the general quantile-regression objective. The quantile $q_{0.5}$ serves as the median prediction, while the learned upper conditional quantiles provide uncertainty-related information.

A.1.3 Bayesian Neural Networks

A BNN places prior distributions on neural-network weights $\boldsymbol{w}$ and approximates their posterior distribution using a variational distribution $q_\phi(\boldsymbol{w})$ Blundell et al. (2015). For a candidate $\boldsymbol{x}$, we draw S posterior weight samples and evaluate the latent function:

$$\boldsymbol{w}^{(s)} \sim q_\phi(\boldsymbol{w}), \qquad f^{(s)}(\boldsymbol{x}) = f(\boldsymbol{x}; \boldsymbol{w}^{(s)}), \quad s = 1, \ldots, S.$$

The posterior sample mean is:

$$\mu(\boldsymbol{x}) = \frac{1}{S} \sum_{s=1}^{S} f^{(s)}(\boldsymbol{x}),$$

and the uncertainty estimates are the sample standard deviation and empirical upper quantiles of these latent-function samples.

A.2 Empirical-Bayes Adjustment: Derivation and Estimation

This section derives the shrinkage weight used in Section 3.2.1 and specifies its out-of-fold (OOF) parameter estimates.

A.2.1 Motivation for the Shrinkage Weight

For a fixed candidate solution $\boldsymbol{x}$ and objective k, let θ_k denote its unknown objective value, $p_k = f_{\text{sur},k}(\boldsymbol{x})$, and $s_k = s_k(\boldsymbol{x})$. Consider the model:

$$\theta_k \sim \mathcal{N}(m_k, \tau_k^2),$$
$$p_k \mid \theta_k \sim \mathcal{N}(\theta_k, c_k s_k^2).$$

Here m_k and τ_k^2 represent the offline objective mean and signal variance, respectively, while c_k is the scaling parameter. For $c_k s_k^2 > 0$ and $\tau_k^2 > 0$, completing the square in the normal posterior yields:

$$\begin{aligned}\mathbb{E}[\theta_k \mid p_k] &= \frac{m_k/\tau_k^2 + p_k/(c_k s_k^2)}{1/\tau_k^2 + 1/(c_k s_k^2)} \\ &= m_k + \frac{\tau_k^2}{\tau_k^2 + c_k s_k^2}(p_k - m_k).\end{aligned}$$

Thus the unmodified shrinkage weight balances the surrogate prediction against the offline objective mean according to their relative variances. The normal likelihood serves as an approximation.

A.2.2 Out-of-Fold Parameter Estimation

Let the offline dataset be $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^N$. We partition it into 5 folds. For each objective, a surrogate trained on the other 4 folds predicts the held-out samples. Denote the resulting prediction and uncertainty estimate for sample i by p_{ik}^{OOF} and s_{ik}^{OOF}, respectively, and let $\mathcal{I}_j$ be the index set of fold j.

Both predictions and real objective values are centered separately within each held-out fold. For $i \in \mathcal{I}_j$, we define:

$$\widetilde{p}_{ik} = p_{ik}^{\text{OOF}} - \frac{1}{|\mathcal{I}_j|}\sum_{r \in \mathcal{I}_j} p_{rk}^{\text{OOF}},$$
$$\widetilde{y}_{ik} = y_{ik} - \frac{1}{|\mathcal{I}_j|}\sum_{r \in \mathcal{I}_j} y_{rk}.$$

Fold centering reduces the effect of mean differences among OOF models. We estimate the offline objective mean and the fold-centered prediction-real covariance as:

$$m_k = \frac{1}{N}\sum_{i=1}^N y_{ik},$$
$$\widehat{\text{Cov}}_k = \frac{1}{N-5}\sum_{i=1}^N \widetilde{p}_{ik}\widetilde{y}_{ik}.$$

The covariance estimate is motivated by the additive-error approximation: $p_k = \theta_k + \varepsilon_k$, with $\mathbb{E}[\varepsilon_k] = 0$ and $\text{Cov}(\theta_k, \varepsilon_k) = 0$. If the real objective equals θ_k, this model gives $\text{Cov}(p_k, y_k) = \text{Cov}(\theta_k, \theta_k) + \text{Cov}(\varepsilon_k, \theta_k) = \text{Var}(\theta_k) + \text{Cov}(\theta_k, \varepsilon_k) = \text{Var}(\theta_k)$. Consequently, we estimate the signal variance using the covariance, with a numerical floor:

$$\widehat{v}_{y,k} = \frac{1}{N-5}\sum_{i=1}^N \widetilde{y}_{ik}^2,$$
$$\epsilon_k = 10^{-12}\max(\widehat{v}_{y,k}, \epsilon_{\text{tiny}}),$$
$$\tau_k^2 = \max(\widehat{\text{Cov}}_k, \epsilon_k),$$

where ϵ_{tiny} denotes the smallest positive normal floating-point number used by the implementation.

Under the same additive-error approximation, the prediction-error variance is: $\mathrm{Var}(p_k) - \mathrm{Cov}(p_k, y_k) = \mathrm{Var}(\theta_k) + \mathrm{Var}(\varepsilon_k) - \mathrm{Var}(\theta_k) = \mathrm{Var}(\varepsilon_k)$. Its OOF analogue is:

$$z_{ik} = \widetilde{p}_{ik}^2 - \widetilde{p}_{ik}\widetilde{y}_{ik},$$
$$\widehat{d}_k = \frac{1}{N-5}\sum_{i=1}^{N} z_{ik},$$
$$\mathrm{SE}(\widehat{d}_k) = \frac{\mathrm{SD}_i(z_{ik})}{\sqrt{N}},$$

where SD_i denotes the sample standard deviation across the N samples, using $N-1$ in its denominator. We use the following non-negative correction:

$$d_k^+ = \begin{cases} \widehat{d}_k \max\left(0, 1 - \dfrac{\mathrm{SE}(\widehat{d}_k)^2}{\widehat{d}_k^2}\right), & \widehat{d}_k > 0, \\ 0, & \widehat{d}_k \le 0. \end{cases}$$

The scaling parameter is then:

$$c_k = \frac{d_k^+}{\max\left(\frac{1}{N}\sum_{i=1}^{N}(s_{ik}^{\mathrm{OOF}})^2, \epsilon_{\mathrm{tiny}}\right)}.$$

This estimation relies on the additive-error assumptions above, which may not hold for biased or misspecified surrogate models.

A.2.3 Variance Clipping and Predictive-Signal Fallback

The clipping threshold is the maximum squared OOF uncertainty estimate:

$$s_{\max,k}^2 = \max_{1 \le i \le N} (s_{ik}^{\mathrm{OOF}})^2.$$

Replacing $s_k^2(\boldsymbol{x})$ by $\min(s_k^2(\boldsymbol{x}), s_{\max,k}^2)$ in the shrinkage weight prevents excessive shrinkage toward m_k. If $c_k \min(s_k^2(\boldsymbol{x}), s_{\max,k}^2) = 0$, the implementation sets $w_k(\boldsymbol{x}) = 1$.

Finally, we calculate the empirical variability estimate as:

$$\mathrm{SE}(\widehat{\mathrm{Cov}}_k) = \frac{\mathrm{SD}_i(\widetilde{p}_{ik}\widetilde{y}_{ik})}{\sqrt{N}}.$$

The EBU adjustment is applied to objective k only when $\widehat{\mathrm{Cov}}_k > \mathrm{SE}(\widehat{\mathrm{Cov}}_k)$. This threshold is a heuristic rather than a statistical significance test, as OOF models share training data across folds.

A.3 Dual-Ranking Survival Selection

Algorithm 1 presents the full survival selection procedure. Non-dominated sorting partitions the candidate solutions into ordered fronts $\mathcal{F}_1, \ldots, \mathcal{F}_J$, where each $\mathcal{F}_j$ contains the candidate solutions assigned to the j-th front.

A.4 Experimental Settings

A.4.1 Surrogate Models

For the proposed surrogate-based methods, one surrogate model is trained independently for each objective. GPR is implemented with GPy using either an automatic relevance determination (ARD) RBF kernel or an ARD Matérn 5/2 kernel. Target normalization is enabled and the kernel and likelihood parameters are optimized during fitting. Predictive uncertainty is computed from the latent-function variance with observation likelihood noise excluded, so the resulting standard deviation represents epistemic uncertainty.

Algorithm 1 Dual-Ranking Survival Selection

Input: Candidate population P; $f_{\text{sur}}(\boldsymbol{x})$ and $f_{\text{UA-sur}}(\boldsymbol{x})$ for all $\boldsymbol{x} \in P$; number of survivors N_s
Output: Next population P_{next}
for each $\boldsymbol{x} \in P$ **do**
    $f_{\text{hyb}}(\boldsymbol{x}) \leftarrow \textit{Concatenate}(f_{\text{sur}}(\boldsymbol{x}), f_{\text{UA-sur}}(\boldsymbol{x}))$
end for
$(\mathcal{F}_1, \ldots, \mathcal{F}_J) \leftarrow \textit{NonDominatedSorting}(P, f_{\text{hyb}})$
$P_{\text{next}} \leftarrow \emptyset$
for $j = 1, \ldots, J$ **do**
    Calculate crowding distances in $\mathcal{F}_j$ using f_{sur}
    if $|P_{\text{next}}| + |\mathcal{F}_j| \leq N_s$ **then**
        $P_{\text{next}} \leftarrow P_{\text{next}} \cup \mathcal{F}_j$
        if $|P_{\text{next}}| = N_s$ **then**
            break
        end if
    else
        $r \leftarrow N_s - |P_{\text{next}}|$
        Select the r solutions from $\mathcal{F}_j$ with the largest crowding distances
        Add the selected solutions to P_{next}
        break
    end if
end for

Quantile regression (QR) is implemented using AutoGluon-Tabular with quantile levels $\{0.5, 0.8, 0.9, 0.95\}$. CatBoost models are excluded from the AutoGluon model pool. The median prediction ($q = 0.5$) is used as the central prediction, while the learned upper quantiles provide uncertainty information. When a standard-deviation representation is required, the $q = 0.9$ prediction is converted to a non-negative Gaussian-equivalent scale using $(q_{0.9} - q_{0.5})/1.2816$.

The Bayesian neural network (BNN) is implemented in Pyro with two fully connected hidden layers of 16 units each and hyperbolic-tangent activations. Independent Gaussian priors with standard deviation 0.5 are placed on all weights and biases. Targets are standardized before training. Variational inference uses an AutoDiagonalNormal guide, stochastic variational inference with the Trace-ELBO objective, and Adam with a learning rate of 10^{-3}. Each model is trained for 50,000 optimization steps. The observation-noise scale is initialized to 10^{-3} and learned under a positivity constraint. At prediction time, 100 posterior samples of the latent function mean are drawn; their mean, standard deviation, and empirical 0.8, 0.9, and 0.95 quantiles are used for prediction and uncertainty estimation. Observation-noise samples are excluded from these uncertainty estimates.

A.4.2 EBU-DR Configuration

For EBU-DR, empirical-Bayes adjustment parameters are estimated using deterministic five-fold out-of-fold predictions on the same selected offline dataset; the test data are not used in this estimation. Under the current configuration, τ^2 uses the floor rule and c uses the dispersion rule, with a positive-part correction applied to the dispersion estimate. EBU-DR performs non-dominated sorting on the concatenated surrogate-predicted and uncertainty-aware objective values, while crowding distances are computed using only the surrogate-predicted objectives. When the empirical-Bayes signal is uninformative, the predictive standard deviation is used as the uncertainty fallback.

A.4.3 Optimization Settings

NSGA-II Deb et al. (2002) is implemented using pymoo Blank & Deb (2020). The population size is 100 and the number of generations is 100, corresponding to exactly 10,000 surrogate evaluations. For each optimization seed, the initial population is selected deterministically from the current offline dataset and is shared across the proposed methods for the same problem, offline dataset, and optimization seed. When $N \geq 100$, 100 samples are selected from a random permutation of the offline dataset. When $N = 50$, a random permutation of the 50 selected samples is repeated once

to form an initial population of 100 individuals. Duplicate elimination is disabled in this case to preserve the fixed population size and evaluation budget.

Simulated Binary Crossover (SBX) Deb et al. (1995) is applied with crossover probability 1.0 and distribution index $\eta = 20$. Polynomial mutation is applied with individual-level probability 1.0, per-variable probability $1/D$, and distribution index $\eta = 20$, where D is the number of decision variables.

A.4.4 Baseline Implementations

Prob-RVEA and Prob-MOEA/D Mazumdar et al. (2022) and TGPR-MO Mazumdar et al. (2023) use adapted implementations of their published probabilistic and tree-guided GPR methods. They fit surrogates on each selected offline subset, with reference-vector and population handling adjusted to return 100 solutions after 100 generations. DDMOEA-GAN Zhang et al. (2024a) trains a WGAN-GP for 2,000 epochs once per offline subset and uses a surrogate ensemble with critic-adjusted NSGA-II. Ridge-regularized quadratic fitting and data-dependent RBF settings are used for small-data training.

The Off-MOO End-to-End, Multiple Models, and COM baselines Xue et al. (2024); Trabucco et al. (2021) use two 2,048-unit hidden layers, 200 training epochs, and a batch size of 128, followed by NSGA-II with a population of 100 for 100 generations. COM's additional 20% Pareto pruning is disabled so that all selected offline samples are retained. PCD Shrestha et al. (2026) and ParetoFlow Yuan et al. (2025) retain their conditional-diffusion and predictor-guided flow-matching sampling mechanisms, respectively, with small-data training adaptations. Both generate 100 final solutions; their generative training is not subject to the evolutionary methods' 10,000-surrogate-evaluation budget. MOBO-Vanilla uses qNEHVI with at most 256 training samples and a 100-candidate proposal batch but is excluded from the main ranking because its results are incomplete.

A.4.5 Evaluation Settings

Surrogate prediction error is evaluated on the independent official test pool. Optimization performance is evaluated using the final population after optimization. For official-pool experiments, objective normalization bounds are computed from the complete official training pool for evaluation only, so HV and IGD+ values remain comparable across sample sizes, offline-data seeds, and methods.

Hypervolume (HV) uses the benchmark-specific raw reference point from the Offline MOO Benchmark Xue et al. (2024), transformed by the same min–max normalization. IGD+ uses the problem Pareto front when available; otherwise, it uses the non-dominated front of the complete official training pool. For each combination of problem, sample size, and method, the configured protocol uses 10 offline-data seeds and 10 optimization seeds.

The raw-space HV reference points used in the experiments are listed in Tables 4 and 5. These points are transformed using the evaluation-only min–max bounds described above before HV is computed.

Table 4: Raw-space HV reference points for the synthetic benchmark problems.

Problem	HV reference point
ZDT1	$(1.10, 8.58)$
ZDT2	$(1.10, 9.59)$
ZDT3	$(1.10, 8.74)$
ZDT4	$(1.10, 300.42)$
ZDT6	$(1.07, 10.27)$
OmniTest	$(2.40, 2.40)$
VLMOP1	$(4.00, 4.00)$
VLMOP2	$(1.10, 1.10)$
VLMOP3	$(9.07, 66.62, 0.23)$
DTLZ1	$(558.21, 552.30, 568.36)$
DTLZ2	$(2.77, 2.78, 2.93)$
DTLZ3	$(1703.72, 1605.54, 1670.48)$
DTLZ4	$(3.03, 2.83, 2.78)$
DTLZ5	$(2.65, 2.61, 2.70)$
DTLZ6	$(9.80, 9.78, 9.78)$
DTLZ7	$(1.10, 1.10, 33.43)$

Table 5: Raw-space HV reference points for the real-world benchmark problems.

Problem	HV reference point
RE21	$(3144.44, 0.05)$
RE22	$(829.08, 2407217.25)$
RE23	$(713710.88, 1288669.78)$
RE24	$(5997.83, 43.67)$
RE25	$(124.79, 10038735.00)$
RE31	$(808.85, 6893375.82, 6793450.00)$
RE32	$(290.66, 16552.46, 388265024.00)$
RE33	$(8.01, 8.84, 2343.30)$
RE34	$(1702.52, 11.68, 0.26)$
RE35	$(7050.79, 1696.67, 397.83)$
RE36	$(10.21, 60.00, 0.97)$
RE37	$(0.99, 0.96, 0.99)$
MO-Portfolio	$(0.29, -0.13)$