# Probabilistic Selection Approaches in Decomposition-based Evolutionary Algorithms for Offline Data-Driven Multiobjective Optimization

Atanu Mazumdar
*University of Jyvaskyla*
*Faculty of Information Technology*
*P.O. Box 35 (Agora), FI-40014 University of Jyvaskyla*
*Finland*

Tinkle Chugh
*Department of Computer Science*
*University of Exeter, UK*

Jussi Hakanen
*University of Jyvaskyla*
*Faculty of Information Technology*
*P.O. Box 35 (Agora), FI-40014 University of Jyvaskyla*
*Finland*

Kaisa Miettinen
*University of Jyvaskyla*
*Faculty of Information Technology*
*P.O. Box 35 (Agora), FI-40014 University of Jyvaskyla*
*Finland*

***Abstract*—In offline data-driven multiobjective optimization, no new data is available during the optimization process. Approximation models, also known as surrogates, are built using the provided offline data. A multiobjective evolutionary algorithm can be utilized to find solutions by using these surrogates. The accuracy of the approximated solutions depends on the surrogates and approximations typically involve uncertainties. In this paper, we propose probabilistic selection approaches that utilize the uncertainty information of the Kriging models (as surrogates) to improve the solution process in offline data-driven multiobjective optimization. These approaches are designed for decomposition-based multiobjective evolutionary algorithms and can, thus, handle a large number of objectives. The proposed approaches were tested on distance-based visualizable test problems and the DTLZ suite. The proposed approaches produced solutions with a greater hypervolume, and a lower root mean squared error compared to generic approaches and a transfer learning approach that do not use uncertainty information.**

***Index Terms*—Kriging, Gaussian processes, metamodelling, surrogate, kernel density estimation, Pareto optimality**

## 1. Introduction

Sometimes, real-world multiobjective optimization problems (MOPs) consisting of conflicting objectives do not have analytical functions or simulation models. Instead, the starting point for optimization is data obtained by, e.g. physical experiments, sensors or real-life processes. The available data can be used to build surrogates (also known as metamodels) that approximate the *underlying* objective functions involved in the phenomenon. Optimization can be performed using these surrogates by embedding them in multiobjective evolutionary algorithms (MOEAs) which have proven to be suitable in solving black-box optimization problems [1]. Data-driven optimization can be categorized as *online* or *offline*. In online data-driven optimization problems, getting more data e.g. by conducting further (expensive) function evaluations is possible, which enables updating the surrogates [2]. However, if no additional data can be acquired during the optimization process, it is known as offline data-driven optimization [3], which is the main focus of this paper.

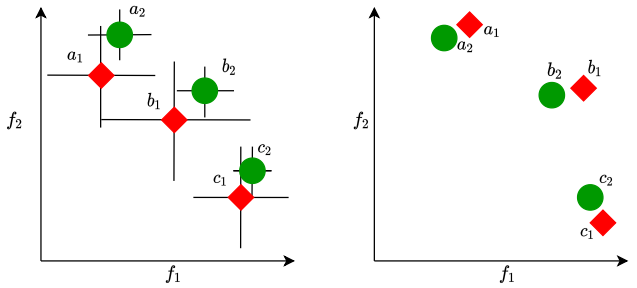



Figure 1: Individuals in the surrogate objective space (left) and underlying objective space (right) for a minimization problem.

Updating the surrogates is not possible in offline data-driven optimization [4]. Thus, while solving an offline MOP, the approximation accuracy and hypervolume of the solutions obtained are entirely dependent on the optimization algorithms used and the surrogates (that involve uncertainty).

In Figure 1, we show an illustration of the objective values and uncertainties of a few individuals approximated by surrogates (e.g. Kriging). For simplicity, we call the objective space of the individuals evaluated with the surrogates and the underlying objective functions as *surrogate objective space* and *underlying objective space*, respectively.

The red individuals dominate the green ones in the surrogate objective space. However, as can be observed, these nondominated individuals also have higher uncertainties. On the right, we show the same individuals evaluated with the underlying objective functions. It can be observed that the green individuals dominate most of the red ones. Thus, while solving offline data-driven MOPs, utilizing just the surrogates' mean approximation can lead to worse solutions. One way to tackle this problem is by using the uncertainty from the surrogates.

Most of the previous works on offline data-driven optimization such as [2], [3], [5] do not consider uncertainty information provided by the surrogates. In [6], optimization is assisted by coarse and fine surrogates. A coarse surrogate is used by the MOEA to find a promising subregion in the search space. Later, the knowledge about good solutions from the coarse search is transferred to the fine search. On the other hand, the works in [7], [8], [9] utilize the approximated uncertainty information provided by Kriging surrogates. The works in [7], [9] use probability of dominance that can be applied to dominance-based MOEAs. The approach in [8] utilizes the approximated uncertainties as additional objective function(s), thereby minimizing the objective values along with the uncertainties in the solutions. However, this approach increases the number of objectives and thus increases the complexity of the optimization problem.

A *generic* approach to solve offline MOPs using MOEA utilises the mean approximations of surrogates as objectives (without uncertainty). The MOEA finds a set of approximated nondominated solutions that represents the trade-offs between the objectives. However, the performance of traditional MOEAs such as MOGA [10], MO-CMA-ES [11], and NSGA-II [12], etc. deteriorates when the number of objectives increases [13], [14], [15]. Decomposition-based MOEAs (e.g. MOEA/D [15], NSGA-III [16] and RVEA [14]) have explicitly been developed to handle a large number of objectives ($> 3$).

In this paper, we propose a probabilistic selection approach and an extended version of it that incorporate uncertainty information in the solution process of offline data-driven MOPs. The proposed approaches utilize techniques such as Monte-Carlo sampling [17] and kernel density estimation (KDE) [18] to estimate the probability of selection criterion in decomposition-based MOEAs. This is particularly advantageous when the closed form of the probability is not available. The proposed approaches are novel in their adaptability or "plug and play" feature for any decomposition-based MOEAs without the requirement of further analytical derivations specific to the MOEA.As an example, we incorporate the proposed probabilistic selection approaches in RVEA and MOEA/D for solving offline MOPs.

The numerical experiments show that the first probabilistic selection approach produces solutions with a better accuracy compared to the generic approach. The second approach proposed is a hybrid selection approach that employs a combination of both the probabilistic and the generic selection approaches. The hybrid approach produces solutions better in hypervolume when compared to the parent approaches. To summarize, the main contributions are:

- Uncertainty information from the surrogates are utilized in the selection process of decomposition-based MOEAs.
- Easy adaptability to any decomposition-based MOEA without any need of analytical derivations.

As we use decomposition-based MOEAs, the proposed approaches are capable of handling a large number of objectives.

The rest of the paper is organized as follows. Basic notations, the background of a generic approach, decomposition-based MOEAs and probabilistic selection are discussed in Section II. The proposed probabilistic and hybrid selection approaches are presented in Section III. Experimental results with analyses are compiled in Section IV. Finally, conclusions and future research perspectives are discussed in Section V.

## 2. Background

In offline data-driven MOPs, there exists no functional form or simulation model which can be accessed during the optimization process. The available (pre-collected) data is the output of a process or phenomenon. As mentioned in the introduction, we refer to the process generating the offline data as *underlying* objective functions. We consider the underlying MOPs of the following form:

$$\begin{aligned} &\text{minimize } \{f_1(\mathbf{x}), \ldots, f_K(\mathbf{x})\}, \\ &\text{subject to } \mathbf{x} \in \Omega, \end{aligned} \tag{1}$$

where $K \geq 2$ is the total number of objectives, and $\Omega$ is the feasible region of the decision space $\mathbb{R}^n$. For a feasible decision vector $\mathbf{x}$, the corresponding objective vector is $\mathbf{f}(\mathbf{x})$, that comprises of the underlying objective (function) values $(f_1(\mathbf{x}), \ldots, f_K(\mathbf{x}))$.

A solution $\mathbf{x}_1 \in \Omega$ dominates another solution $\mathbf{x}_2 \in \Omega$ if $f_k(\mathbf{x}_1) \leq f_k(\mathbf{x}_2)$ for all $k = 1, \ldots, K$ and $f_k(\mathbf{x}_1) < f_k(\mathbf{x}_2)$ for at least one $k = 1, \ldots, K$. A solution of an MOP is nondominated if it is not dominated by any other feasible solution. An MOEA typically produces solutions that are nondominated within the set of solutions it has found. The solutions of (1) that are nondominated in $\Omega$ are also called Pareto optimal solutions. In what follows, we refer to solutions of MOEAs as approximated Pareto optimal ones. The set of solutions in the objective space is called the Pareto front and the corresponding set of decision vectors is the Pareto set.

### 2.1. Generic Offline Data-Driven Multiobjective Optimization

The generic approach to solve offline MOPs is shown in Figure 2. As described in [3], [4], the solution process can be split into three stages which are (a) data collection, (b)

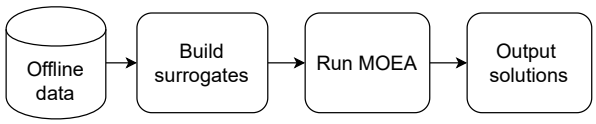



Figure 2: Flowchart of a generic offline data-driven multi-objective optimization approach.

formulating the MOP and surrogate building, and (c) optimization using an MOEA. The initial step of data collection may include pre-processing if necessary. Surrogates are then built using the provided offline data. Some of the popular surrogates that are used for solving offline data-driven MOPs are Kriging [19], neural networks [20], and support vector machines [20]. Finally, an MOEA is used to solve the MOP with the surrogates as objectives.

## 2.2. Decomposition-based MOEAs in Brief

Decomposition-based MOEAs are designed to solve MOPs with more than three objectives [21], [22]. In general, they decompose the problem into a number of single objective subproblems using scalarizing functions (e.g. MOEA/D [15]) or multiple MOPs (e.g. MOEA/D-M2M [23], NSGAIII [16] and RVEA [14]). In this paper, we use RVEA and MOEA/D as two decomposition-based algorithms and apply our proposed approaches in them. These algorithms start by creating a set of $N$ uniformly distributed unit reference vectors (or weight vectors) $\mathbf{v_j}$ $(j = 1, \ldots, N)$. The population of individuals is $P$ and the objective vectors for the individuals are $F = \{\mathbf{f}_1, \ldots, \mathbf{f}_{|P|}\}$ consisting of $|P|$ individuals. The $i^{th}$ individual in $P$ is denoted by $I_i$. The vector of minimum objective function values present in the given population is $\mathbf{z}^{min} = (z_1^{min}, \ldots, z_K^{min})$.

**2.2.1. MOEA/D in brief.** MOEA/D [15] performs search in the neighbourhood of each reference vector, it updates the population sequentially. In each neighbourhood, the offspring population is generated using crossover and mutation, which is then compared with the parent population. A selection criterion, e.g. PBI or Tchebycheff scalarizing function, is used to select the population for the next generation. In this article, we use PBI as the selection criterion and evaluate it for $\mathbf{x}$ as:

$$g_{\mathbf{x}}^{PBI} = d_1 + \rho d_2, \tag{2}$$

where parameter $\rho$ is the penalty term that balances between convergence and diversity, $d_1 = ||(\mathbf{z}^{min} - \mathbf{f}(\mathbf{x}))^T \mathbf{v}_j||/||\mathbf{v}_j||$ and $d_2 = ||\mathbf{f}(\mathbf{x}) - (\mathbf{z}^{min} - d_1 \mathbf{v}_j)||$, respectively. Here $\mathbf{v}_j$ is the $j^{th}$ reference vector in the neighbourhood. MOEA/D updates solutions in their neighbourhood by checking if $g_{\mathbf{x}'}^{PBI} \leq g_{\mathbf{x}_j}^{PBI}$, then set $\mathbf{x}_j = \mathbf{x}'$ and $\mathbf{f}(\mathbf{x}_j) = \mathbf{f}(\mathbf{x}')$. Here, $\mathbf{x}'$ is the offspring and $\mathbf{x}_j$ is the $j^{th}$ solution in the neighbourhood.

**2.2.2. RVEA in brief.** The RVEA algorithm first translates the objective vectors as $\mathbf{f}_i' = \mathbf{f}_i - \mathbf{z}^{min}$, where $i = 1, \ldots, |P|$. It then splits the population into subpopulations by assigning individuals to reference vectors by measuring the cosine between the reference vector and the translated objective vector. The cosine value between the $j^{th}$ reference vector $\mathbf{v}_j$ and the $i^{th}$ translated objective vector $\mathbf{f}_i'$ is given by:

$$cos\ \theta_{i,j} = \frac{\mathbf{f}_i' \cdot \mathbf{v}_j}{\|\mathbf{f}_i'\|}, \tag{3}$$

where $\|\mathbf{f}_i'\|$ is the Euclidean norm. An individual $I_i$ is included in the $z^{th}$ subpopulation $\bar{P}_z$ if it has the lowest angle $\theta_{i,j}$ between $\mathbf{f}_i'$ and $\mathbf{v}_z$ (or highest $cos\,\theta_{i,j}$ value). The index of the $z^{th}$ reference vector to which individual $I_i$ is assigned is:

$$I_i | z = \underset{j \in \{1, \ldots N\}}{\operatorname{argmax}}\ cos\ \theta_{i,j}. \tag{4}$$

After the individuals are assigned to subpopulations, RVEA selects the $z^{th}$ individual from each subpopulation, which has the minimum APD between the $i^{th}$ individual and the $j^{th}$ reference vector according to:

$$I_z | z = \underset{i \in \{1, \ldots, |\bar{P}_j|\}}{\operatorname{argmin}}\ d_{i,j}, \tag{5}$$

where APD (or $d_{i,j}$) is defined as,

$$d_{i,j} = (1 + P(\theta_{i,j})) \cdot ||\mathbf{f}_i'||. \tag{6}$$

Here $P(\theta_{i,j}) = K \cdot (t/t_{max})^{\alpha} \cdot \theta_{i,j}/\gamma_{\mathbf{v}_j}$ is the penalty function depending on $\theta_{i,j}$, and $\gamma_{\mathbf{v}_j} = \min_{i \in 1, \ldots, N, i \neq j} \langle \mathbf{v}_i, \mathbf{v}_j \rangle$, is the smallest angle between reference vector $\mathbf{v}_j$ and the other reference vectors. Here $t$ is the generation counter, $t_{max}$ is the maximum number of generations and $\alpha$ controls the rate of change of $P(\theta_{i,j})$. For more details, see [14].

## 2.3. Probabilistic Selection in Single Objective Optimization

We here provide an overview of probabilistic selection in single objective optimization, which is then further extended to solve offline data-driven MOPs. Let us consider a single objective offline data-driven minimization problem where the given data may have noise due to e.g., experimental or measurement error. The total uncertainty is due to the noise in the data (that can be estimated by Kriging surrogate) and the uncertainty in the approximation. Due to this, an individual with worse (greater) underlying objective values may be selected.

For example, in Figure 3, we have individuals A and B with uncertain objective values. These two individuals have a normally distributed probability density function (PDF). The red star shows an example of a random sample $y$ drawn from $\mathrm{PDF}_A$. When a random sample is drawn from $\mathrm{PDF}_B$, we may observe a smaller value than $y$ thus making us select the individual with a worse objective value. The total probability of selecting the wrong individual B over A by observing a specific sample is the total area in the shaded region under $\mathrm{PDF}_B$, or cumulative density function (CDF) of B (denoted by $\mathrm{CDF}_B(y)$). The probability of drawing a random sample $y$ is $\mathrm{PDF}_A(y)$. Thus the total probability of

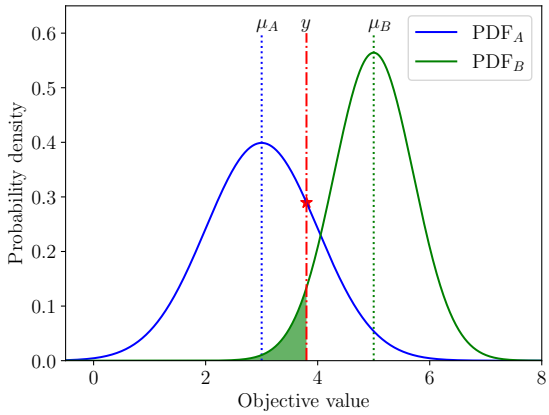



Figure 3: Probability of choosing the individual with worse underlying objective value in selection between two individuals with uncertain objective values (approximated by the surrogate) for a single objective minimization problem.

a random sample drawn from $\text{PDF}_B$ being smaller than $y$ is $\text{PDF}_A(y) \cdot \text{CDF}_B(y)$. The probability of wrongly choosing B over A when the underlying objective value of A is smaller than that of B according to [7] is:

$$P_{wrong}(A > B) = \int_{-\infty}^{\infty} \text{PDF}_A(A-y) \cdot \text{CDF}_B((A-y) > (B-y))\ dy, \quad (7)$$

We can replace $\text{CDF}_B$ as an integral of $\text{PDF}_B$ as:

$$P_{wrong}(A > B) = \int_{-\infty}^{\infty} (\text{PDF}_A(y) \cdot \int_{-\infty}^{y} \text{PDF}_B(\mu) d\mu)\ dy. \quad (8)$$

The work in [7] used the following equation for comparing a set of individuals with uncertain objective values and ranking them based on their probabilities:

$$R_i = \sum_{n=1}^{|P|} P_{wrong}(I_n > I_i) - 0.5, \quad (9)$$

where $R_i$ is the ranking score given to the $i^{th}$ individual $I_i$. The total number of individuals to be compared is $|P|$, and $P_{wrong}(I_n > I_i)$ is the probability of making a wrong decision in selection such that the fitness of $I_i$ is smaller than the fitness of $I_n$. A value of 0.5 is subtracted from the ranking function as $P_{wrong}(I_i > I_i)$ is always 0.5. The individual with the best fitness value will have the smallest rank or has the smallest probability of making the wrong selection.

## 3. Probabilistic decomposition-based MOEAs

We utilize uncertainty in decomposition-based MOEAs and propose approaches to solve offline data-driven MOPs. Hence, the original selection process in decomposition-based MOEAs has to be modified to utilize the uncertainty approximated by the surrogates. This can be done by utilizing (7) and (8) to formulate a probabilistic selection criterion specific to the MOEA. The probability of the selection criterion can be computed analytically. However, this is quite complex in decomposition-based MOEAs, and the selection criterion has to be tailor-made for every variant of MOEA. Next, we describe our plug and play probabilistic approaches.

We summarize the generalized steps of the proposed probabilistic approach in Algorithm 1. We start with offline data of size $N_D$. Next, we build a Kriging surrogate for each underlying objective and initialize $N$ uniformly distributed unit reference vectors [24]. For initializing the population we use the provided offline data set. We generate offspring using crossover and mutation in the neighbourhood (a fixed number in MOEA/D and the maximum number of reference vectors in RVEA) and use Kriging models for approximating the objective values of the offspring. As the approximation distribution of Kriging surrogate is Gaussian, the multivariate PDF [19] for $I_i$ is:

$$\text{PDF}_{I_i} = \prod_{k=1}^{K} \frac{1}{\hat{\sigma}_{i,k}\sqrt{2\pi}} \exp\left(-\frac{(f_k - \hat{f}_{i,k})^2}{2\hat{\sigma}_{i,k}^2}\right), \quad (10)$$

where $\hat{f}_{i,k}$ is the approximated $k^{th}$ objective function value for the $i^{th}$ individual with $\hat{\sigma}_{i,k}$ as its standard deviation.

We draw $S$ samples using Monte-Carlo sampling [17] from the distribution in (10) for the $i^{th}$ individual. Individuals are then assigned to sub-populations in a probabilistic manner depending on the decomposition-based MOEA. The selection criterion is then calculated for all the generated samples of objective values. In the next step, we apply Kernel density estimation (KDE) [18] to approximate the distribution of the selection criterion depending on the decomposition-based MOEA. However, we may skip this step if the closed form distribution of the selection criterion is available (e.g. weighted sum and Tchebycheff as shown in the supplementary material). We then use these estimated PDFs to select individuals in a probabilistic way. The details of KDE are provided in the supplementary material. The reference vectors are then adapted after a certain number of generations (or function evaluations) to obtain a uniformly distributed set of solutions [14]. The stopping criterion is the maximum number of function evaluations performed with surrogates.

### 3.1. Probabilistic Selection in RVEA

The two major modifications to incorporate uncertainties in approximated objective values in RVEA are a) the assignment of individuals to reference vectors and b) the selection of an individual using probabilistic APD (steps 10 and 11, respectively, in Algorithm 1).

**3.1.1. Probabilistic Assigning to Reference Vectors.** As mentioned in Section II, assigning individuals to respective reference vectors in RVEA depends on the objective values as in (3). However, when the objective values (provided by the Kriging models) have uncertainties, assigning individuals can not be deterministic. Hence, in step 10 we perform a

**Algorithm 1:** Probabilistic decomposition-based MOEA

**Input:** Offline data of size $N_D$; $N$ = number of reference vectors; $FE_{max}$ = maximum number of function evaluations using Kriging surrogates; $S$ = number of samples to be used for estimating the distributions

**Output:** Approximated solutions

1 Build Kriging surrogates for each objective using the given offline data
2 Use the given data as the initial population; initialize the number of function evaluations $FE = 0$
3 Create a set of uniformly distributed unit reference vectors $V_0$ of size $N$
4 Find the neighbourhood for each unit reference vector
5 **while** $FE < FE_{max}$ **do**
6  Perform crossover and mutation on population and generate offspring
7  Evaluate the individuals using the Kriging surrogates and combine the parents and offspring
8  Update $FE = FE + |P_{\text{offspring}}|$
9  Draw $S$ samples using Monte-Carlo from the distribution approximated by the surrogates
10  Perform algorithm specific sub-population assigning
11  Perform algorithm specific probabilistic selection
12 end

probabilistic assigning of individuals to reference vectors by using the distribution of the approximated objective values.

As explained previously, we draw $S$ samples using Monte-Carlo sampling from the distribution in (10) for the $i^{th}$ individual in the current population. The vector of samples is used to calculate $cos\,\theta_{i,j,l}$ for the $i^{th}$ individual and $j^{th}$ reference vector using (3), where $l = 1, \ldots, S$ is the sample number. These samples can then be used to estimate the PDF of $cos\,\theta_{i,j}$ using KDE. We use (9) for ranking the PDFs of $cos\,\theta_{i,j}$ that gives us the rank $R_{i,j}$. We use the following modification of (4) to include an individual to a subpopulation:

$$\bar{P}_z = \left\{ I_i | z = \underset{j \in \{1,\ldots,N\}}{\text{argmax}} \; R_{i,j} \right\}, \tag{11}$$

where

$$R_{i,j} = \sum_{n=1}^{N} P_{wrong}(\, cos\,\theta_{i,n} > \; cos\,\theta_{i,j}) - 0.5 \tag{12}$$

and $\bar{P}_z$ is the $z^{th}$ subpopulation and $R_{i,j}$ is the probabilistic rank of assigning an individual to a subpopulation.

Computing $P_{wrong}$ in (8) is computationally expensive as it involves double integration. Hence, calculating $R_{i,j}$ in (12) for all the individuals becomes computationally expensive with a complexity of $O(N \cdot |P|)$. This is especially high when the numbers of individuals and reference vectors are large. To reduce the computation time, we calculate how many samples out of $S$ for every individual are assigned to different reference vectors instead of performing numerical integration. By such a voting mechanism, the $i^{th}$ individual is assigned to the reference vector to which most samples are assigned.

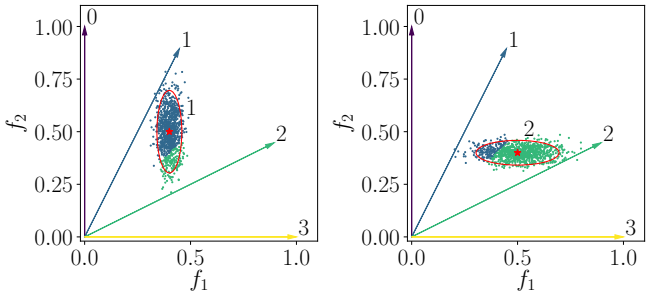



Figure 4: Distribution of samples drawn for two individuals with different mean objective values (indicated by the red star) and standard deviations (indicated by ellipses). The colour code indicates the reference vector a sample drawn from the PDF of the individual is assigned to. The reference vector the individual is assigned to is indicated by the number.

We provide an illustration of the probabilistic assigning of individuals to reference vectors in Figure 4. It shows the samples drawn for two individuals in the objective space for the approximated distribution and the corresponding reference vectors they are assigned to. Samples assigned to a reference vector are colour coded. The red stars show the mean of the distributions of objective values for the individuals as approximated by the surrogate. The ellipses show the distributions of objective values for two standard deviations. It can be observed in sub-figure (a) that most of the samples are 'blue' or get assigned to reference vector '1'. Hence we can assign the individual to reference vector '1'. However, in sub-figure (b) when the distributions of objective values are changed, the individual gets assigned to reference vector '2'.

**3.1.2. Probabilistic Angle Penalized Distance.** After assigning individuals to reference vectors, we select one individual from every subpopulation in step 11 of Algorithm 1. Given a subpopulation and its associated individuals with their PDFs, we utilize the samples for each individual and calculate their APD values using (6). These samples of APD values are used to estimate the distribution of APD for every individual, $\text{PDF}_{d_{i,j}}$ using KDE. In this work, we used Gaussian kernel and adopted the Silvermann's rule [25] for selecting the bandwidth parameter that controls the smoothness of the estimated distribution. The estimated

PDFs of APD are ranked by modifying (5) utilizing (9) as:

$$P_{nextgen} = \left\{ I_z | z = \underset{i \in \{1,\ldots,|\bar{P}_j|\}}{\operatorname{argmin}} R'_{i,j} \right\}, \quad (13)$$

where

$$R'_{i,j} = \sum_{n=1}^{|\bar{P}_j|} P_{wrong}(d_{n,j} > d_{i,j}) - 0.5. \quad (14)$$

The rank of the $i^{th}$ individual in the subpopulation $\bar{P}_j$ is $R'_{i,j}$. The $z^{th}$ individual $I_z$ is selected from subpopulation $\bar{P}_j$, where $j = 1, \ldots, N$, for population of the next generation $P_{nextgen}$.

To find $R'_{i,j}$ in (14), a pairwise comparison between $\text{PDF}_{d_{i,j}}$ has to be performed. Thus, the computation cost of performing the probabilistic selection is $O(|\bar{P}_j|^2)$, where $|\bar{P}_j|$ is the number of individuals in the $j^{th}$ subpopulation. The overall computation cost becomes high due to the double integral involved while finding $P_{wrong}$ between $\text{PDF}_{d_{i,j}}$ of two individuals. Besides, there is an additional computation cost involved while performing the KDE of $\text{PDF}_{d_{i,j}}$.

We propose an approach to compute $P_{wrong}$ in an efficient way. As we know from (13), the calculation of the rank matrix $R'_{i,j}$ in (14) is a pairwise comparison. Thus $P_{wrong}$ between an APD distribution with itself is always 0.5. Also, the computation needs to be done just once for the same pair. We can alter the calculation of $P_{wrong}$ as follows:

$$P_{wrong}(d_{n,j} > d_{i,j}) = \begin{cases} 0.5 \text{ if } n = i, \\ 1 - P_{wrong}(d_{i,j} > d_{n,j}) \text{ if } n > i. \end{cases} \quad (15)$$

The double integration in (8) with the inner integral responsible for calculating the CDF from the approximated PDF contributes the most to the computation cost. Instead, we can use a coarse approximation of the CDF by computing the empirical CDF from the APD samples which would reduce the computation cost. To compute $P_{wrong}$ in (8), the lower limit during integration can be changed to zero instead of $-\infty$. This is because APD can never attain a value below zero, and thus the PDF should be adjusted to estimate the probability density for APD values lower than zero. An illustration of the estimated PDF and empirical CDF calculated from the APD samples is shown in Figure 5(a). To further reduce the computation cost, $P_{wrong}$ values for all the subpopulation individuals are computed in parallel. Applying all the proposed cost reduction approaches reduced the computation cost for computing the rank $R'_{i,j}$ for every generation.

### 3.2. Probabilistic Selection in MOEA/D

In MOEA/D, the assignment of solutions for each reference vectors is performed by defining the neighborhood of each vector [15]. Thus, step 10 in Algorithm 1 can be skipped. Next we demonstrate how to implement probabilistic selection with PBI as selection criterion (in step 11 of Algorithm 1).

Similar to the probabilistic selection in RVEA, we first draw $S$ samples from the approximated distribution of objectives for the individuals in the neighbourhood and the offspring. The vector of sampled objective values is used to calculate PBI samples for the $j^{th}$ individual in the neighbourhood and the offspring that we call as $g^{PBI}_{\mathbf{x}_{j,l}}$ and $g^{PBI}_{\mathbf{x}'_l}$, respectively, where $l = 1, \ldots, S$. Next, we approximate the PDF of $g^{PBI}_{\mathbf{x}_j}$ and $g^{PBI}_{\mathbf{x}'}$ using KDE. Again, similar to the probabilistic RVEA, we use Gaussian kernel and Silvermann's rule to select the bandwidth parameter. Since we are comparing PDFs of PBI, we need to modify the update operation in generic MOEA/D and calculate the $P_{wrong}$ utilizing (7). We update the solutions in the neighbourhood by checking if $P_{wrong}(g^{PBI}_{\mathbf{x}'} \leq g^{PBI}_{\mathbf{x}_j}) < 0.5$, then set $\mathbf{x}_j = \mathbf{x}'$ and $\mathbf{f}(\mathbf{x}_j) = \mathbf{f}(\mathbf{x}')$.

It has to be noted that the comparison of PDFs of PBI is one to many, whereas for PDFs of APD its pairwise. In Figure 5(b), we show the estimated PDF of PBI samples for one of the individuals and the empirical CDF.

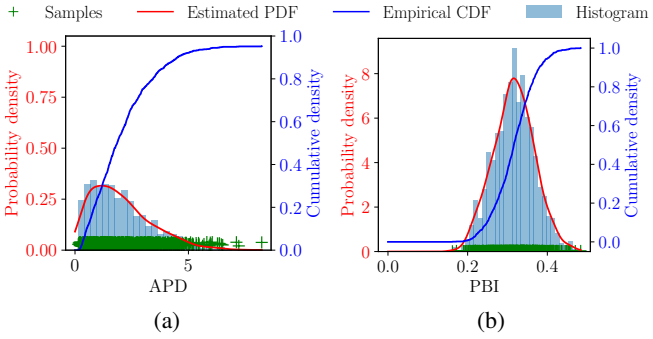



Figure 5: Samples drawn, histogram, estimated PDF and empirical CDF of the selection criterion for (a) RVEA (APD) and (b) MOEA/D (PBI) for one individual.

We illustrate the potential of the proposed probabilistic selection approach for a bi-objective minimization problem in Figure 6. It shows different individuals with uncertain objective values, with error bars representing the 95% confidence interval of the approximated distributions. The numbers represent the reference vector / sub-population the individuals are assigned to. Green individuals are the ones selected from a sub-population. It can be observed that for the individuals assigned to reference vector '2', the original selection criterion (generic approach) selects the individual that is better in objective values even though it has a much higher uncertainty. On the other hand, the probabilistic approach selects the individuals that are worse in terms of objective values but have comparatively lower uncertainties. For the individuals assigned to reference vector '0', both the individuals have the same approximated mean values. However, the probabilistic approach selects the one with lower uncertainties. For individuals assigned to reference vector '3' or in its neighbourhood, both the individuals have

the same uncertainties with slightly different objective values. One can see that both the generic and the probabilistic approach select the same individual with a better value of the selection criterion.

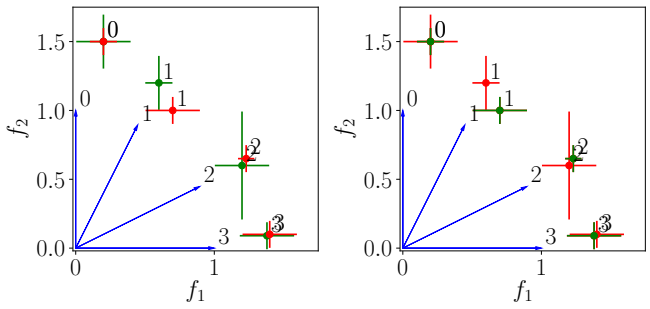



Figure 6: Selection of individuals using (a) generic approach and (b) probabilistic approach for a minimization problem. Error bars show the 95% confidence interval of the distribution of objective values approximated by the surrogates. Colours 'green' and 'red' show the individuals that are selected and not selected by the approaches, respectively.

### 3.3. Hybrid of Probabilistic and Generic Approaches

The proposed probabilistic approach tends to select individuals with better objective values and lower uncertainties (or high approximation accuracy). On the other hand, the generic approach, without incorporating any uncertainty, produces solutions with better hypervolume. Hence, a hybrid of the two approaches can provide the benefits of both and produce a set of solutions with a wider range of uncertainties and objective values. This is especially advantageous for decision making due to the wider choices it provides [26].

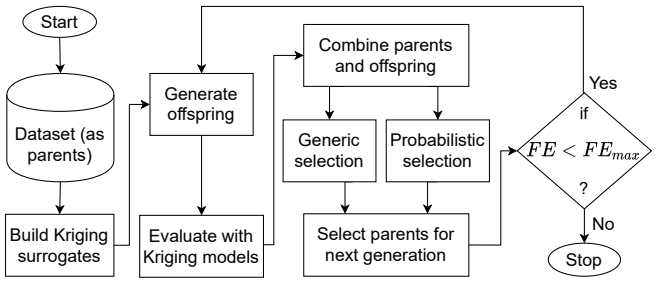



Figure 7: Flowchart for the hybrid approach.

A flowchart of the hybrid approach is shown in Figure 7. In the proposed hybrid approach, we select individuals based on both the criteria. An equal proportion of solutions from both the generic and probabilistic selection criteria are selected. This choice also helped in avoiding the need of introducing extra parameters. The redundant copies of the individuals (that were selected by both the approaches) are removed and the entire selected population is used for further crossover and mutation steps. It has to be noted that all the solutions obtained by the generic and the probabilistic approach (that are equal in number) are used, and no further parameters are required.

## 4. Experimental Results

In this section, we demonstrate the potential of the proposed approaches embedded in RVEA and MOEA/D by solving distance-based multiobjective visualizable test problems (DBMOPP) [27] and DTLZ [28] test problems with different numbers of objectives. DBMOPP problems have certain advantages over traditional benchmark suites. First, we can simultaneously visualize the solutions in both decision and objective spaces. Second, we can visualize the search behaviour with the number of function evaluations. Third, these problems do not have a limitation of having the same values of many decision variables on the Pareto set as in the DTLZ problems.

### Experimental setup

- Benchmark problems: Two sets of DBMOPP problems denoted as P1 and P2 (details are in the supplementary material) utilizing the code in [29]. Test results with DTLZ suite are provided in the supplementary material.
- Number of objectives ($K$): 2-10
- Number of decision variables ($n$): 10
- Termination ($FE_{max}$): 40000 function evaluations with surrogates.
- Kriging parameters: Scikit-learn python library [30] for Kriging with Gaussian kernel and BFGS [31] to maximize the marginal likelihood.
- Approach specific parameters: Number of samples for Monte-Carlo sampling $S = 1000$. For KDE we used Gaussian kernel with a bandwidth parameter set by Silvermann's rule (for details, see supplementary material).
- Other algorithms: We compared the proposed probabilistic approaches for RVEA and MOEA/D with three generic approaches that use the approximated values (posterior mean in this case) denoted as generic (Gen-RVEA and Gen-MOEA/D) and transfer learning (TL) approach [6] and the initial samples (Init). The probabilistic approaches for RVEA and MOEA are denoted by Prob-RVEA and Prob-MOEA/D, respectively. The hybrid approaches for RVEA and MOEA/D are denoted by Hyb-RVEA and Hyb-MOEA/D, respectively.
- MOEA parameter settings: The reference vectors were generated by simplex lattice design [24]. The number of reference vectors was varied with the number of objectives as in [24]. All the approaches except TL used reference vector adaptation after every 10th generation as in [14] for all test instances. We used simulated binary crossover with distribution index 30 and probability 1.0, polynomial mutation

with distribution index 20 and probability $1/n$. For APD we set $\alpha = 2$. For MOEA/D the neighbourhood size was 20 and $\theta$ in PBI varied from 0 to 500 with function evaluations.

- Number of independent runs for each instance: 31
- Performance metrics: We used hypervolume and RMSE as metrics to measure the performance of the tested approaches. Further details regarding metrics and reference points are provide in the supplementary material.
- Size of the initial data set ($N_D$): 109
- Sampling of the initial data set: Latin hypercube sampling (LHS) and multivariate normal sampling (MVNS). In MVNS sampling, the objectives were considered to be independent with mean at the midpoint of the decision space (0 for P1 and P2). The variance of the sampling distribution was set to 0.1 for all the objectives. Similarly, the mean of the distribution for DTLZ instances was set to 0.5 with variance of 0.1 for all objectives.

All the approaches were implemented in Python utilizing the DESDEO framework (desdeo.it.jyu.fi) [1]. For the TL approach, we used the Matlab implementation in [6].

It should be noted that we evaluated the approximated solutions obtained with different approaches with underlying objective function values to calculate hypervolume and RMSE. This is only for experimentation and such evaluations may not be possible while solving real-life offline MOPs. We refer to the hypervolume of the solutions evaluated with the surrogates as *surrogate hypervolume* for simplicity.

A pairwise Wilcoxon significance test [32] was conducted between the different approaches and the calculated p-values were later Bonferroni corrected. The threshold $\alpha = 0.05$ was considered for rejecting the null hypothesis. The overall ranking of the approaches was done by a scoring system where the approach considered as the alternate hypothesis is given a score +1 when it is significantly better than the null hypothesis. A score of zero is given if the alternate approach is not significantly better or worse than the null hypothesis. If the alternate approach is significantly worse than the null hypothesis, it gets a score -1. The sum of the scores of the hypothesis testings is used for ranking all the approaches in a descending order of the score.

We show the median hypervolumes and RMSEs (along with the standard deviations in different runs) for a few DBMOPP test instances in Table 1. The items in bold represent the best performing approaches. We also show the heatmap for all the tested DBMOPP instances comparing hypervolume and RMSE in Figure 8. The rankings of the approaches are colour-coded from best (yellow) to worst (purple) using the 'viridis' colourmap. It can be observed that most of the yellow colours are around Prob-RVEA, Hyb-RVEA and Prob-MOEA/D.

1. Python source code can be accessed from https://github.com/industrial-optimization-group/offline_data_driven_moea

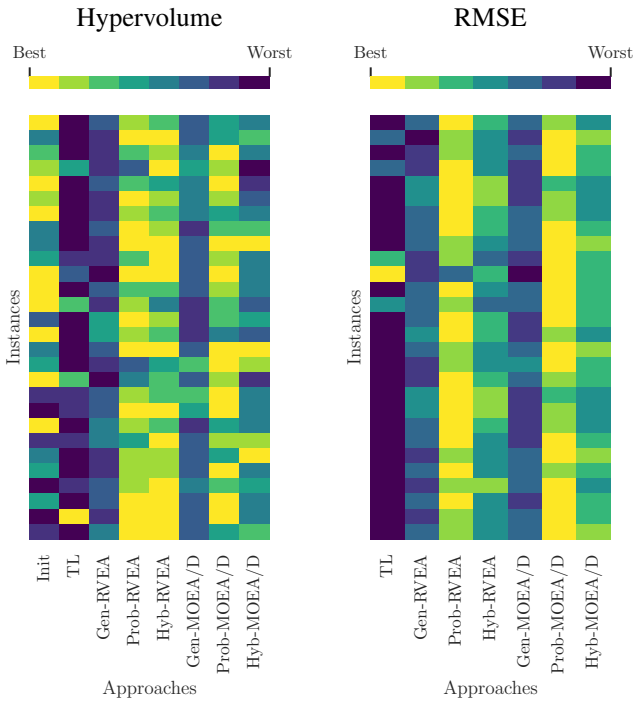



Figure 8: Heatmaps of: (a) hypervolume and (b) RMSE of solutions obtained by initial sampling, transfer learning and the generic, probabilistic and hybrid approaches for RVEA and MOEA/D respectively for DBMOPP problem instances.

As we can observe, Hyb-RVEA and Prob-RVEA performed best in terms of hypervolume followed by Prob-MOEA/D. In terms of RMSE, Prob-MOEA/D performed the best with Prob-RVEA coming second. For certain test instances, the generic approaches and TL produced worse solutions (both in hypervolume and RMSE) compared to the initial sampling. This is because they do not consider uncertainty in approximations when selecting solutions and thus converged far from the Pareto front. Detailed results on other DBMOPP and DTLZ instances are provided in the supplementary material.

Overall, the probabilistic approaches outperformed their generic counterparts, TL and initial sampling in both hypervolume and RMSE. However, we found that Hyb-MOEA/D did not perform better than Prob-MOEA/D. This is because of the extremely poor performance of Gen-MOEA/D in terms of both hypervolume and RMSE.

We also show the progress or search behaviour of different approaches of the runs with median hypervolume values in Figure 9. The first column shows the problem instance and the initial samples (red '+' showing the non-dominated ones). The problem instance in the top row has five objectives, and in the bottom row eight objectives with two disconnected Pareto sets. The black circles with centers denoted by '+' are the Pareto set and dots on the circle with different colors represent the points to which the objectives or distances are minimized. In other words, if solutions are

TABLE 1: Hypervolume (HV) and RMSE for a few DBMOPP test instances.

| Sampling | Problem | K | Metric | Init. | TL | Gen-RVEA | Prob-RVEA | Hyb-RVEA | Gen-MOEA/D | Prob-MOEA/D | Hyb-MOEA/D |
|---|---|---|---|---|---|---|---|---|---|---|---|
| LHS | P1 | 8 | HV | **9.09E+05** (2.65E+04) | 5.22E+05 (1.65E+05) | 5.98E+05 (1.00E+05) | 7.38E+05 (1.01E+05) | 6.88E+05 (9.63E+04) | 6.17E+05 (9.67E+04) | 6.52E+05 (9.70E+04) | 6.18E+05 (8.83E+04) |
| | | | RMSE | — | 1.76E+00 (3.92E-01) | 1.11E+00 (4.01E-01) | **1.59E+00** (3.69E-01) | 1.55E+00 (3.87E-01) | 1.73E+00 (4.16E-01) | 1.53E+00 (3.93E-01) | 1.61E+00 (3.84E-01) |
| | P2 | 8 | HV | 7.33E+05 (7.11E+04) | 3.86E+05 (1.24E+05) | 7.77E+05 (5.27E+04) | **8.34E+05** (3.88E+04) | **8.16E+05** (4.24E+04) | 7.24E+05 (3.33E+04) | 7.78E+05 (3.90E+04) | 7.73E+05 (4.01E+04) |
| | | | RMSE | — | 1.56E+00 (2.76E-01) | 1.34E+00 (3.48E-01) | **1.48E+00** (2.96E-01) | 1.39E+00 (2.23E-01) | 1.67E+00 (2.81E-01) | **1.34E+00** (3.21E-01) | 1.46E+00 (3.23E-01) |
| MVNS | P1 | 6 | HV | 6.13E+03 (4.31E+02) | 7.22E+03 (1.82E-12) | 7.73E+03 (3.35E+02) | **8.62E+03** (3.23E+02) | **8.52E+03** (3.24E+02) | 7.78E+03 (4.50E+02) | **8.71E+03** (3.76E+02) | 7.78E+03 (2.53E+02) |
| | | | RMSE | — | 1.85E+00 (1.31E-01) | 1.77E+00 (1.57E-01) | **1.81E+00** (1.19E-01) | 1.80E+00 (1.53E-01) | 1.89E+00 (1.19E-01) | 1.79E+00 (1.30E-01) | 1.85E+00 (1.22E-01) |
| | P2 | 10 | HV | 4.52E+07 (2.92E+06) | 3.37E+07 (1.10E+07) | 8.05E+07 (8.61E+06) | 8.62E+07 (9.34E+06) | **9.14E+07** (4.53E+06) | 6.96E+07 (7.29E+06) | 8.85E+07 (2.74E+06) | 8.70E+07 (4.60E+06) |
| | | | RMSE | — | 1.11E+00 (3.50E-01) | 5.00E-01 (3.59E-01) | **9.16E-01** (3.22E-01) | 9.43E-01 (3.64E-01) | 1.29E+00 (3.48E-01) | **4.50E-01** (3.00E-01) | 8.81E-01 (3.35E-01) |

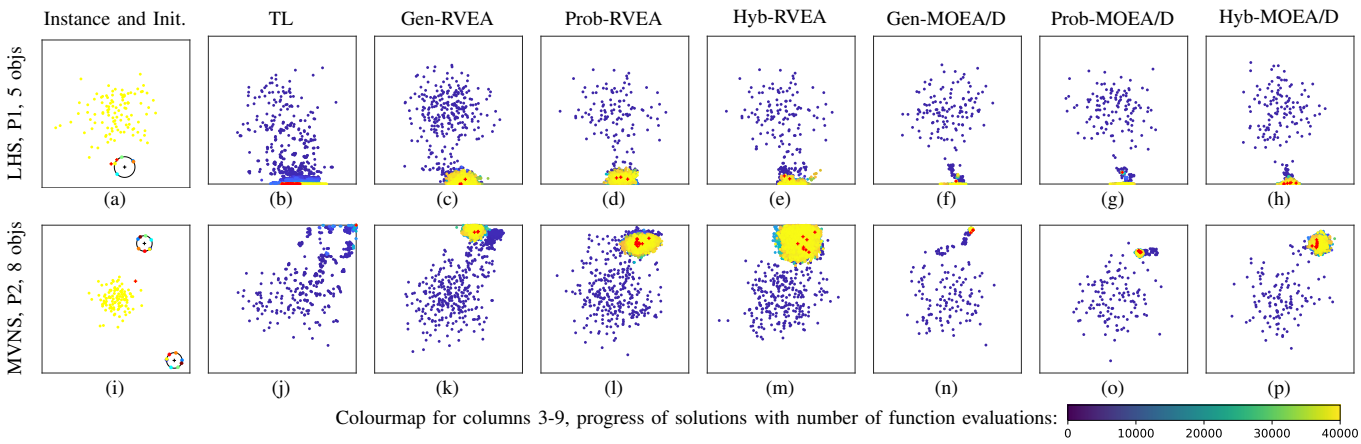


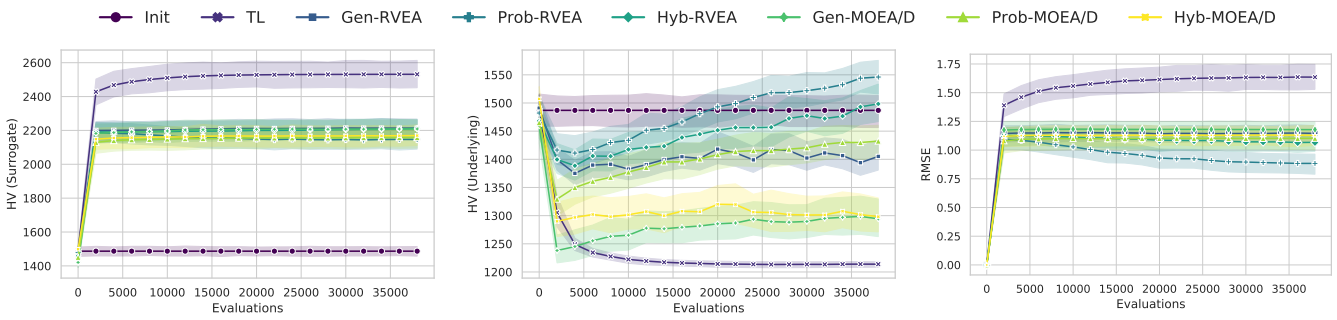

Figure 9: Progress of the solution process with median hypervolume values. DBMOPP P1 with $K = 5$ with LHS and P2 with $K = 8$ with MVNS. The colour code represents the number of function evaluations during the solution process.

on or in the circle, they are on the Pareto front. Therefore, with these visualizations, we can see how close the solutions from an approach can get to the Pareto front. The next seven columns show the progress of the solutions by evaluating with the underlying objectives for the different approaches tested. The color of the solutions represent the function evaluations count, and the nondominated solutions of the last generations are shown in red '+'. For visualizing the solution of the tested DBMOPP problem instances, a 10-dimensional decision space is projected to 2-dimensional.

It can be observed that Prob-RVEA in (d) and (l), Hyb-RVEA in (e) and (m), Prob-MOEA/D in (g) and (o) and Hyb-MOEA/D in (h) and (p) converged much closer to the Pareto front. However, the solutions for the Gen-RVEA in (c) and (k), Gen-MOEA/D in (f) and (n), and TL in (b) and (j) converged further away from the Pareto front. One can also see that the nondominated solutions in the initial sampling in (a) and (i) are much closer to the Pareto front compared to the generic approaches and TL. We also observed that all approaches failed to get solutions closer to both disconnected Pareto sets in the bottom row. This is because of the lack of adequate offline data for solving such MOPs.

In Figure 10, we show the surrogate hypervolume, hypervolume in the underlying objective space and RMSE after every 1000 function evaluations. As can be seen, the probabilistic and hybrid approaches improved the hypervolume in the underlying objective space with function evaluations. However, the generic approaches and TL could not further improve the hypervolume. On the contrary, the surrogate hypervolume for TL improved with function evaluations and the probabilistic approaches it gradually decreased. This is because the probabilistic approaches reject solutions with better objective values if they have high uncertainties. The RMSE increased for all the approaches in the initial function evaluations. However, in the later function evaluations, it gradually reduced for the probabilistic approaches (especially for Prob-RVEA). The final RMSE in TL was the worst.

The performances of the generic approaches and TL were poor because they did not consider uncertainty in the solutions during the optimization process. Hence, the




(a) (b) (c)

Figure 10: Progress of the solution process with median hypervolume values. DBMOPP P1 with $K = 5$ and LHS showing the variation of hypervolume (HV) in surrogate and underlying objective spaces, and RMSE with function evaluations.

solutions, when evaluated with the underlying objectives, produced worse objective values compared to what was observed in the surrogate objective space. TL performed the worst because the sparse search adopted in the approach led to faster convergence in the surrogate objective space but far from the Pareto front.

## 5. Conclusions

We have proposed two probabilistic selection approaches for solving offline data-driven MOPs. These approaches are designed for decomposition-based MOEAs and utilize uncertainty in the approximations of Kriging surrogates. The first approach estimates the distribution of selection criterion embedding in a decomposition-based MOEA. The second one is a hybrid of the first approach and a generic approach (that does not use any uncertainty in the approximations). We demonstrated the potential of the proposed approaches with RVEA and MOEA/D, that also showed its adaptability. For benchmarking, we used several distance-based multi-objective visualizable test problems and DTLZ problems. We used different sampling techniques and numbers of objectives.

Considering the accuracy of the solutions is often neglected in offline data-driven optimization. The proposed probabilistic approaches are more focused on improving the accuracy of the solutions. A detailed analysis of quality measures (hypervolume and RMSE), visualization of the solution process and comparison with existing approaches clearly showed the advantages of the proposed approaches. For the problem instances tested, we can conclude that depending on the indicators used, MOEAs and problem characteristics, both probabilistic and hybrid approaches have their strengths. However, if more accurate solutions are required, the probabilistic approach should be preferred.

In future, we plan to test the proposed approaches in other decomposition-based MOEAs with different normalization techniques. We will also test on real-world problems and develop approaches to handle constraints.

## References


[1] Y. Jin, "Surrogate-assisted evolutionary computation: Recent advances and future challenges," *Swarm and Evol. Comput.*, vol. 1, pp. 61–70, 2011.

[2] H. Wang, Y. Jin, and J. O. Jansen, "Data-driven surrogate-assisted multiobjective evolutionary optimization of a trauma system," *IEEE Trans. Evol. Comput.*, vol. 20, pp. 939–952, 2016.

[3] H. Wang, Y. Jin, C. Sun, and J. Doherty, "Offline data-driven evolutionary optimization using selective surrogate ensembles," *IEEE Trans. Evol. Comput.*, vol. 23, pp. 203–216, 2019.

[4] Y. Jin, H. Wang, T. Chugh, D. Guo, and K. Miettinen, "Data-driven evolutionary optimization: An overview and case studies," *IEEE Trans. Evol. Comput.*, vol. 23, pp. 442–458, 2019.

[5] H. Wang and Y. Jin, "A random forest-assisted evolutionary algorithm for data-driven constrained multiobjective combinatorial optimization of trauma systems," *IEEE Trans. Cyber.*, vol. 50, pp. 536–549, 2020.

[6] C. Yang, J. Ding, Y. Jin, and T. Chai, "Offline data-driven multi-objective optimization: Knowledge transfer between surrogates and generation of final solutions," *IEEE Trans. Evol. Comput.*, vol. 24, pp. 409–423, 2020.

[7] E. J. Hughes, "Evolutionary multi-objective ranking with uncertainty and noise," in *Evolutionary Multi-Criterion Optimization, Proc.*, E. Zitzler, L. Thiele, K. Deb, C. A. Coello Coello, and D. Corne, Eds. Springer, 2001, pp. 329–343.

[8] A. Mazumdar, T. Chugh, K. Miettinen, and M. López-Ibáñez, "On dealing with uncertainties from Kriging models in offline data-driven evolutionary multiobjective optimization," in *Evolutionary Multi-Criterion Optimization, Proc.*, K. Deb, E. Goodman, C. A. Coello Coello, K. Klamroth, K. Miettinen, S. Mostaghim, and P. Reed, Eds. Springer, 2019, pp. 463–474.

[9] A. A. M. Rahat, C. Wang, R. M. Everson, and J. E. Fieldsend, "Data-driven multi-objective optimisation of coal-fired boiler combustion systems," *Applied Energy*, vol. 229, pp. 446–458, 2018.

[10] C. M. Fonseca and P. J. Fleming, "Genetic algorithms for multiobjective optimization: Formulation discussion and generalization," in *Proceedings of the 5th International Conference on Genetic Algorithms*. Morgan Kaufmann Publishers Inc., 1993, pp. 416–423.

[11] T. Voß, N. Hansen, and C. Igel, "Recombination for learning strategy parameters in the MO-CMA-ES," in *Evolutionary Multi-Criterion Optimization, Proc.* Springer, 2009, pp. 155–168.

[12] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, pp. 182–197, 2002.

[13] L. C. T. Bezerra, M. López-Ibáñez, and T. Stützle, "A large-scale experimental evaluation of high-performing multi- and many-objective evolutionary algorithms," *Evol. Comput.*, vol. 26, pp. 621–656, 2018.

[14] R. Cheng, Y. Jin, M. Olhofer, and B. Sendhoff, "A reference vector guided evolutionary algorithm for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 20, pp. 773–791, 2016.

[15] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 11, pp. 712–731, 2007.

[16] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints," *IEEE Trans. Evol. Comput.*, vol. 18, pp. 577–601, 2014.

[17] N. Metropolis, "The beginning of the Monte Carlo method," *Los Alamos Science*, pp. 125—130, 1987.

[18] M. Rosenblatt, "Remarks on some nonparametric estimates of a density function," *Ann. Math. Stat.*, vol. 27, pp. 832–837, 1956.

[19] A. Forrester, A. Sobester, and A. Keane, *Engineering Design via Surrogate Modelling*. John Wiley & Sons, 2008.

[20] C. Bishop, *Pattern Recognition and Machine Learning*. Springer-Verlag New York, 2006.

[21] K. Li, "Decomposition multi-objective evolutionary optimization: From state-of-the-art to future opportunities," 2021.

[22] A. Trivedi, D. Srinivasan, K. Sanyal, and A. Ghosh, "A survey of multiobjective evolutionary algorithms based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 21, pp. 440–462, 2017.

[23] H.-L. Liu, F. Gu, and Q. Zhang, "Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems," *IEEE Trans. Evol. Comput.*, vol. 18, pp. 450–455, 2014.

[24] R. Cheng, Y. Jin, K. Narukawa, and B. Sendhoff, "A multiobjective evolutionary algorithm using gaussian process-based inverse modeling," *IEEE Trans. Evol. Comput.*, vol. 19, pp. 838–856, 2015.

[25] B. Silverman, *Density Estimation for Statistics and Data Analysis*. London: Chapman Hall/CRC, 1986.

[26] A. Mazumdar, T. Chugh, J. Hakanen, and K. Miettinen, "An interactive framework for offline data-driven multiobjective optimization," in *Bioinspired Optimization Methods and Their Applications, Proc.*, B. Filipič, E. Minisci, and M. Vasile, Eds. Springer, 2020, pp. 97–109.

[27] J. E. Fieldsend, T. Chugh, R. Allmendinger, and K. Miettinen, "A feature rich distance-based many-objective visualisable test problem generator," in *Proceedings of the Genetic and Evolutionary Computation Conference*. ACM, 2019, pp. 541–549.

[28] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable test problems for evolutionary multiobjective optimization," in *Evolutionary Multiobjective Optimization: Theoretical Advances and Applications*, A. Abraham, L. Jain, and R. Goldberg, Eds. Springer, 2005, pp. 105–145.

[29] J. E. Fieldsend, T. Chugh, R. Allmendinger, and K. Miettinen, "fieldsend/DBMOPP_generator," 2019, accessed June 15, 2021. [Online]. Available: "https://github.com/fieldsend/DBMOPP_generator"

[30] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.

[31] M. Avriel, *Nonlinear Programming: Analysis and Methods*. Dover Publishing, 2003.

[32] W. Frank, "Individual comparisons by ranking methods," *Biometrics*, vol. 1, pp. 80–83, 1945.