On Dealing with Uncertainties from Kriging Models in Offline Data-driven Evolutionary Multiobjective Optimization

Atanu Mazumdar[1], Tinkle Chugh[2], Kaisa Miettinen[1], and Manuel López-Ibáñez[3]

[1] University of Jyvaskyla, Faculty of Information Technology, P.O. Box 35 (Agora), FI-40014 University of Jyvaskyla, Finland

[2] Department of Computer Science, University of Exeter, UK

[3] Alliance Manchester Business School, University of Manchester, UK

Abstract. Many works on surrogate-assisted evolutionary multiobjective optimization have been devoted to problems where function evaluations are time-consuming (e.g., based on simulations). In many real-life optimization problems, mathematical or simulation models are not always available and, instead, we only have data from experiments, measurements or sensors. In such cases, optimization is to be performed on surrogate models built on the data available. The main challenge there is to fit an accurate surrogate model and to obtain meaningful solutions. We apply Kriging as a surrogate model and utilize corresponding uncertainty information in different ways during the optimization process. We discuss experimental results obtained on benchmark multiobjective optimization problems with different sampling techniques and numbers of objectives. The results show the effect of different ways of utilizing uncertainty information on the quality of solutions.

Keywords: Machine learning · Gaussian process · Pareto optimality · Metamodelling · Surrogate

1 Introduction

Sometimes in real applications, multiple conflicting objectives should be optimized, but there is no mathematical or simulation model of the objectives involved. Instead, there is data, e.g., obtained via physical experiments. In such cases, surrogate models can be built using the given data and optimization is then performed with the surrogate models. In the literature, surrogate models such as Kriging [8], neural networks [18] and support vector regression [16] have been typically used for solving computationally expensive optimization problems [6,10]. If we may conduct new (expensive) function evaluations when needed, this process is called online data-driven optimization [20]. When we do not have access to additional data during the optimization, we call it *offline data-driven optimization* [11].

In using surrogate models, the main challenge is to manage the models for improving convergence and diversity without too much sacrifice in the accuracy of models. In online data-driven optimization problems, an infill criterion [6] is maximized or minimized for updating the models iteratively during the optimization process. However, this is not applicable for offline data-driven optimization when no further data is available during the optimization process.So far, little research has been conducted on solving optimization problems, where no new data is available for managing the surrogates [4,11,20]. In such case, the quality of the solutions obtained after using the surrogate models is entirely dependent on the accuracy of the models and optimizer used.

When solving an offline data-driven problem with multiple conflicting objectives, one can fit models using all the data available for each objective function. Then an evolutionary multiobjective optimization (EMO) algorithm can be used on these models to find a set of approximated nondominated solutions. Essentially, in that case, an offline data-driven multiobjective optimization problem (MOP) can be divided into two major parts: model building and using an EMO algorithm.

Some surrogate models, like Kriging, provide uncertainty information (or standard deviation) about the predicted values. A low standard deviation implies that the actual objective function value has a higher chance of being close to the predicted value (though the actual function may remain unknown and the only information is the data available). Therefore, one possible way to improve the accuracy of the model is to utilize uncertainty in the fitted model as an additional objective to be optimized.

In this article, we study different ways to deal with the uncertainty information provided by the Kriging models in offline data-driven multiobjective optimization. Moreover, we consider the effect of using different initial sampling techniques on some benchmark test problems. In this study, we simulate offline problems by generating data for problems with known optimal solutions to be able to analyze the results. The results show the effect of utilizing uncertainty information in the quality of solutions.

The rest of this article is organized as follows. We summarize the basic concepts of data-driven optimization and Kriging model in Section 2. In Section 3, we present different approaches of incorporating uncertainty information in the optimization problem and present and analyze the results in Section 4. Finally, we draw conclusions in Section 5.

2 Background

2.1 Generic Offline Data-Driven EMO

We consider MOPs of the following form :

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

with k (≥ 2) objective functions and the feasible set S is a subset of the decision space $\mathbb{R}^n$. For any feasible decision vector $\mathbf{x}$ we have a corresponding objective vector $f(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_k(\mathbf{x}))$.

MOPs that are offline in nature can generally be solved by the approach given in Fig. 1. In what follows, we refer to it as a *generic approach*. As described in [11,21], the solution process can be split into three major components: (1) data collection, (2) model building and management, and (3) EMO method utilized. The collection of data may also incorporate data pre-processing, if it is required. Once the data has been obtained, the objectives and constraints of the MOP are formulated. The next stage is to build surrogate models (also known as meta-models) e.g. for each objective function using the available data. Finally, an EMO method is used to find nondominated solutions utilizing the surrogates as objective functions. As objectives to be optimized in (1) we have for $i = 1, \ldots, k$ the predicted means $\hat{f}_i$ of the surrogate of objective f_i and our objective vector is denoted by:

$$\hat{\mathbf{f}} = (\hat{f}_1(\mathbf{x}), \ldots, \hat{f}_k(\mathbf{x})). \tag{2}$$

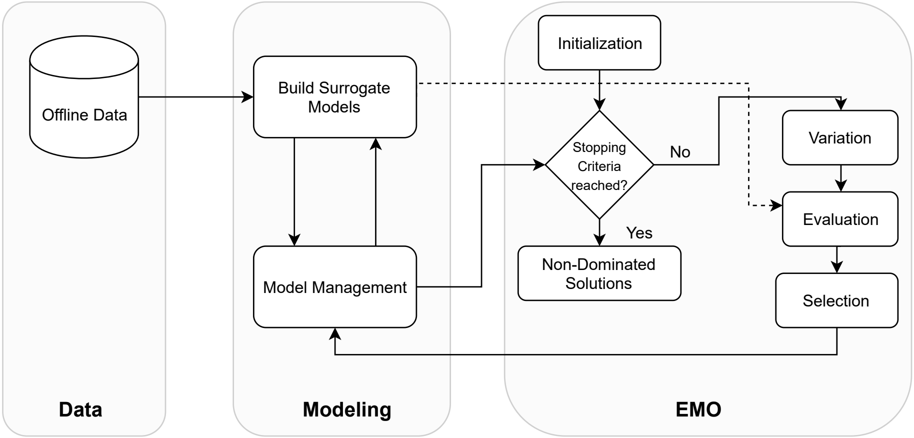

Fig. 1. Flowchart of a generic offline data-driven evolutionary multiobjective optimization approach.

Selecting proper surrogate models is a challenging task in model management. In online data-driven EMO, the quality of the surrogate models can be accessed and updated as new data becomes available during the optimization process. However, for offline data-driven EMO this is not possible. It becomes even more challenging with the data being noisy [22], skewed [23], time-varying [2] or heterogeneous [3]. Thus, it is crucial to build, before optimization, surrogates that are as good approximations as possible of the "true" objective functions. One way to improve the accuracy of the surrogates is to enhance the quality of the data. In this research, our consideration is on a general level and we do not go into the characteristics of the data.

In offline data-driven EMO, the possible ways to improve the accuracy of the surrogate models are to have an effective data pre-processing for noise removal [4], creating synthetic data [23], transferring knowledge [15] or applying advanced machine leaning techniques [19,20]. However, it is quite possible that the surrogate models are not good representations of the true objectives. It may even happen that the solutions obtained are actually worse than the data used for fitting the models.

2.2 Kriging

Kriging or Gaussian process regression has been widely used as a surrogate model for solving expensive optimization problems [6]. The main advantage of using Kriging is its ability to provide uncertainty information of the predicted values. Given a Kriging model, the approximated mean value y^* and its variance s^2 for a sample (or decision variable value) $\mathbf{x}^*$ are as follows:

$$y^* = \mathbf{k}(\mathbf{x}^*, \mathbf{X}) K(\mathbf{X}, \mathbf{X})^{-1} \mathbf{y}, \tag{3}$$

$$s^2 = \mathbf{k}(\mathbf{x}^*, \mathbf{x}^*) - K(\mathbf{x}^*, \mathbf{X}) K(\mathbf{X}, \mathbf{X})^{-1} K(\mathbf{X}, \mathbf{x}^*), \tag{4}$$

where $\mathbf{X} \in \mathbb{R}^{N_I \times n}$ is the matrix of the given data with N_I items with n decision variables, $\mathbf{y} \in \mathbb{R}^{N_I}$ is the vector of given objective values corresponding to some decision vector, $K(\mathbf{X}, \mathbf{X})$ is the covariance matrix of $\mathbf{X}$ and $\mathbf{k}(\mathbf{x}^*, \mathbf{X})$ is a vector of covariances between $\mathbf{x}^*$ and $\mathbf{X}$. For more details about Kriging, see [17].

3 Approaches to Incorporate Uncertainty

As new data cannot be obtained in offline data-driven optimization, it is difficult to update the surrogates and enhance their accuracy. One approach is to build a very accurate surrogate model before the optimization process. Another possible approach is to provide a suitable metric in addition to final solutions after the optimization process, which can be used to measure the accuracy of solutions obtained. This approach can be beneficial when the surrogate models cannot provide a very exact representation of the true objective functions. One such instance can be when the data consists of optimal solutions. In such a case, the surrogate might not be a good representation of the actual objectives, which might lead to degraded final solutions. Providing a set of solutions together with the uncertainty information of predicted final solutions can be helpful in the decision making process.

As previously discussed, the two major components in offline data-driven optimization are building a surrogate model and using an EMO algorithm. In this research we have limited ourselves by focusing on a few variations of the optimization problem which try to minimize the uncertainty in the final solutions. As shown in Figure 2, the uncertainties in the predicted value of the Kriging models are utilized as additional objective functions. By considering uncertainties in this way, the EMO method tries to minimize the predicted mean values from the fitted Kriging models by subsequently minimizing the standard deviations in the prediction. Thus, the final set of nondominated solutions will consist of solutions with different levels of uncertainty.

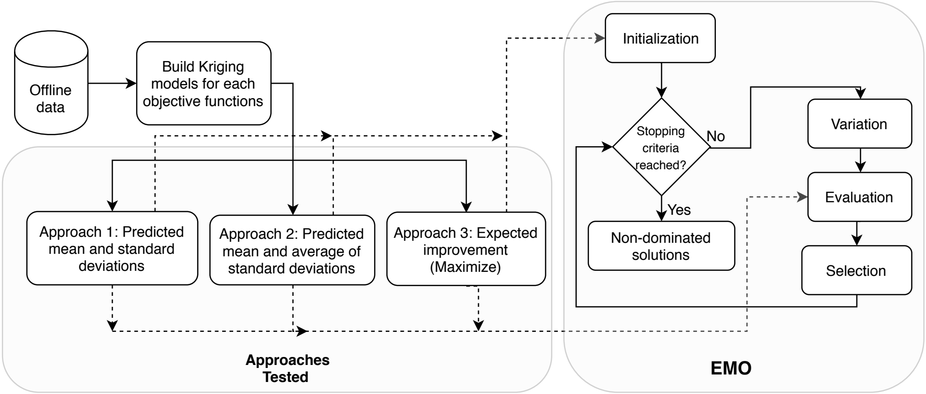
Offline data
Build Kriging models for each objective functions
Approach 1: Predicted mean and standard deviations
Approach 2: Predicted mean and average of standard deviations
Approach 3: Expected improvement (Maximize)
Approaches Tested
Initialization
Stopping criteria reached?
No
Yes
Variation
Evaluation
Selection
Non-dominated solutions
EMO

Fig. 2. Flowchart of offline data-driven optimization with uncertainty.

We have tested three different approaches for utilizing uncertainties in the optimization. Approach 1 uses all the standard deviations given by each surrogate model as additional objectives. The resulting objective vector in Approach 1 is:

$$\hat{\mathbf{f}} = (\hat{f}_1(\mathbf{x}), \ldots, \hat{f}_k(\mathbf{x}), s_1(\mathbf{x}), \ldots, s_k(\mathbf{x})), \tag{5}$$

where $\hat{f}_i(\mathbf{x})$ and $s_i(\mathbf{x})$ and are the predicted mean and the standard deviation values for the i^{th} objective. Final solutions are obtained by performing a non-dominated sort on the archive of predicted solutions (predicted mean values and standard deviations) stored while optimization. It might be possible that the solutions have different uncertainties for different objectives. We double the number of objectives which may increase the complexity of solving the resulting optimization problem.

Approach 2 utilizes the average of the standard deviations given by each of the surrogate models as an additional objective and the resulting objective vector is:

$$\hat{\mathbf{f}} = (\hat{f}_1(\mathbf{x}), \ldots, \hat{f}_k(\mathbf{x}), \bar{s}(\mathbf{x})), \tag{6}$$

where $\bar{s}(\mathbf{x})$ is the average of the standard deviations from Kriging models built for each objective function. This method has fewer objectives when compared to Approach 1, however, either of the approaches provide solutions with a range of uncertainty values. Both Approaches 1 and 2 can provide an option for filtering solutions based on the uncertainty information.

Algorithm 1 Uncertainties as additional objective functions

```
Input: $k$ Kriging models, one for each objective function and an empty archive
Output: Final nondominated approximated solutions from the archive
Generate parent population
while Stopping criteria are not reached do
    Generate offspring with crossover and mutation
    Evaluate offspring using Kriging models and get the objective function values
        of either Eqs. (4), (5) or (6)
    Combine offspring population with parent population
    Select parents for the next generation
    Store parents in the archive
Perform nondominated sorting of solutions in the archive
```

Approach 3 utilizes the expected improvement (EI) [12] for every surrogate model as objectives to be optimized by the EMO algorithm, see, e.g. [9]. Expected improvement can be expressed as $\mathrm{EI}(\mathbf{x}) = (f_{min} - \hat{f}(\mathbf{x}))\Phi\left(\frac{f_{min}-\hat{f}(\mathbf{x})}{s(\mathbf{x})}\right) + s(\mathbf{x})\phi\left(\frac{f_{min}-\hat{f}(\mathbf{x})}{s(\mathbf{x})}\right)$, where $\phi(\cdot)$ and $\Phi(\cdot)$ are the standard normal density and distribution function respectively, and f_{min} is a k-dimensional vector, where the i^{th} component represents the best values of the i^{th} objective function in the given data. The objective vector in this case is:

$$\hat{\mathbf{f}} = (\mathrm{EI}_1(\mathbf{x}), \ldots, \mathrm{EI}_k(\mathbf{x})), \tag{7}$$

where $\mathrm{EI}_i(\mathbf{x})$ is the expected improvement value for the i^{th} objective. The EI criterion takes the predicted mean value and the standard deviation into account.

Now we have introduced three approaches for incorporating uncertainty information. Algorithm 1 shows the process of applying any of them in the offline optimization process, where k is the number of objectives and we can use the maximum number of evaluations using surrogate models as a stopping criterion.

4 Experimental Results

We compare the three different approaches to each other and also to a generic approach (as (2) in Subsection 2.1), using test problems DTLZ2, DTLZ4–DTLZ7 with 2, 3 and 5 objectives. As said, we generate data for these problems and fit Kriging models there. The dimension of the decision variable space n is fixed to 10.

The size of the data set used is 109 (corresponds to the $11n - 1$ [5,13,24]). The sampling techniques for creating the data sets were *Latin hypercube sampling* (LHS), *uniform random sampling* and a special case of sampling which we call *optimal-random sampling*. In the latter, 50% of the data are nondominated solutions and the remaining 50% are uniform random samples. This kind of hypothetical sampling might resemble a special case where most of the samples in the given data set are close to optimal, and thus the optimization process could no longer improve the solutions further. However, in such a scenario the offline optimization technique should not compute final solutions which are worse than the provided samples. A total of 31 independent runs from each sampling were performed for each case.

We used indicator based evolutionary algorithm (IBEA) [25] as the EMO method as it has been demonstrated to perform well in [1] even for problems with a higher number of objectives. The selection criterion was $I_{\epsilon+}$ (Step 6 in Algorithm 1) with κ parameter values 0.51, 0.87 and 0.48 for $k = 2, 3$ and 5, respectively, and κ value of 0.5 for any other number of objectives. The population size was 100 and the maximum number of function evaluations was 40 000 according to [1]. We used Matlab implementation of Kriging models with first order polynomial functions and a Gaussian kernel function.

For measuring the performance of different approaches, we first performed a nondominated sort on the archive (also including the additional objective(s)). These nondominated solutions were then evaluated with the real objective function. After obtaining their true objective function values, dominated solutions were removed producing the final nondominated set. For comparing the quality of solutions for all the approaches, inverted generational distance (IGD) metric was utilized with 5000 points in the reference set for all problems.

Table 1 shows the comparison between the mean and standard deviation values of the IGD for all the three approaches and the generic approach. It was observed that Approaches 1 and 2 performed better than the generic approach for LHS and uniform random sampling for all the problems with various numbers of objectives with the exception of DTLZ6 and DTLZ7. However, while using optimal-random sampling, Approaches 1 and 2 performed better than the generic approach for DTLZ2, DTLZ4-5 and better for DTLZ6 and DTLZ7 for few of the objectives. Approach 3 did not produce good results for any of the problems, objectives or sampling technique.

Adding uncertainties as additional objectives pose a major problem in explaining the effect of optimization as the fitness landscape of the uncertainties is mostly unknown. A possible explanation that no noticeable performance improvement is observed in DTLZ6 when using Approaches 1 and 2 is because the problem consists has a non-uniform (or biased) [7] degenerated Pareto front. Adding additional uncertainty objectives makes the problem even harder to solve and fewer nondominated solutions are obtained. For DTLZ7, a possible explanation for the worse performance of Approaches 1 and 2 is that the objective functions are completely separable [14]. Thus, the additional objectives added by Approaches 1 and 2 only make the problem more difficult than the generic approach.

For optimal-random sampling the advantage of Approaches 1 and 2 was clearly visible. Despite the initial sampling including also nondominated solutions, the generic approach failed to provide good solutions. This is because the surrogate models do not provide a perfect representation of the true objectives. While utilizing EIs as objectives in Approach 3, the solutions were actually worse (comparing mean IGD values) for most of the cases. This is because EI tries to balance between convergence and diversity. Therefore, it can select a solution with a high uncertainty for achieving its goal.

Figure 3 shows the root mean square error (RMSE) of the final solutions obtained by different approaches with LHS sampling on problems with two objectives. It can be observed that the solutions obtained by Approaches 1 and 2 are more accurate in most of the cases. This means that using uncertainty as additional objective(s) helps to find solutions with a low approximation error. Therefore, using uncertainty in the optimization process can be considered as an advantage in solving an offline data-driven EMO problem where there is no possibility for updating the surrogate models. An illustration of solutions obtained after evaluating them with real objectives for the DTLZ2 problem with LHS and optimal-random sampling is shown in Figure 4. Due to space limitations, further analysis is available at http://www.mit.jyu.fi/optgroup/extramaterial.html as additional material. The performance of the proposed approaches on other test problems (i.e., DTLZ1, DTLZ3, WFG1-WFG3, WFG5 and WFG9) can also be found at the above-mentioned website.

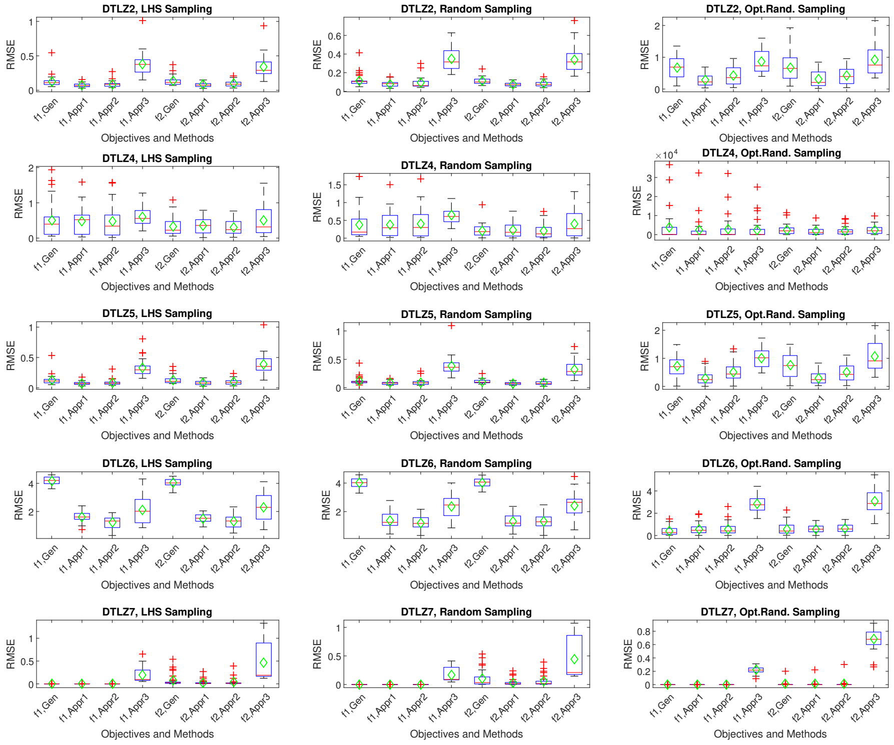

Fig. 3. RMSE of the final solutions for bi-objective problems. Here f1 and f2 are the objectives and "Gen","Appr1","Appr2" and "Appr3" are the generic and Approaches 1, 2 and 3, respectively. Opt.Rand is optimal-random sampling.

Table 1. Means and standard deviations of IGD values of the final archive, evaluated on the true objective functions, obtained by each approach, for various problems and sampling techniques. (Best values are in bold)

Sampling	**Problems**	**k**	**Generic**		**Approach 1**		**Approach 2**		**Approach 3**	
			Mean	Std.Dev.	Mean	Std.Dev.	Mean	Std.Dev.	Mean	Std.Dev.
LHS	DTLZ2	2	0.0989	0.1260	**0.0722**	**0.0431**	0.0770	0.0651	0.3377	0.0477
		3	0.2027	0.0910	0.1787	0.0530	**0.1665**	0.0539	0.3471	**0.0365**
		5	0.2708	0.0873	0.2689	**0.0343**	**0.2574**	0.0396	0.3993	0.0395
	DTLZ4	2	0.6311	**0.1619**	**0.3951**	0.1935	0.4919	0.1852	0.6467	0.2098
		3	0.7306	0.2021	**0.5309**	0.1413	0.5867	0.1467	0.7166	**0.1162**
		5	0.6929	0.0766	**0.5640**	0.0653	0.6062	0.0545	0.7173	**0.0514**
	DTLZ5	2	0.1030	0.1326	0.1032	0.0905	**0.0814**	**0.0570**	0.3716	0.0580
		3	0.1191	0.0982	**0.0684**	**0.0315**	0.0701	0.0452	0.2676	0.0388
		5	0.0934	0.0606	**0.0655**	**0.0277**	0.0805	0.0453	0.1486	0.0387
	DTLZ6	2	**0.1570**	**0.1078**	1.6188	0.7635	2.4518	0.5797	3.5210	1.1369
		3	**0.9871**	**0.2737**	1.7564	0.7308	1.5561	0.7159	3.2847	1.1907
		5	**0.8207**	**0.2158**	2.3859	0.4822	1.3725	0.3734	2.8157	1.0211
	DTLZ7	2	**0.0023**	**0.0049**	0.0292	0.0095	0.0095	0.0086	0.6157	0.1767
		3	**0.0549**	**0.0120**	0.1791	0.1721	0.0956	0.1449	0.6529	0.1016
		5	**0.2800**	**0.0541**	0.5254	0.2175	0.3675	0.1234	0.7169	0.0888
Random	DTLZ2	2	0.0947	0.0893	0.0879	0.0468	**0.0828**	0.0493	0.3673	**0.0395**
		3	0.2315	0.0712	0.1907	0.0534	**0.1692**	**0.0316**	0.3591	0.0433
		5	0.2843	0.0790	0.2593	**0.0268**	**0.2514**	0.0335	0.4188	0.0289
	DTLZ4	2	0.5986	0.1857	**0.4461**	0.1850	0.4665	**0.1735**	0.4935	0.2243
		3	0.7885	0.1465	**0.5354**	0.1474	0.5682	**0.1320**	0.7680	0.1544
		5	0.7064	0.1731	**0.5487**	0.1021	0.6034	0.1127	0.7391	**0.0697**
	DTLZ5	2	0.1144	0.1211	0.0949	0.0495	**0.0889**	0.0506	0.3590	**0.0481**
		3	0.1114	0.0367	**0.0610**	0.0291	0.0615	**0.0283**	0.2823	0.0350
		5	0.0644	0.0447	**0.0498**	**0.0169**	0.0542	0.0254	0.1521	0.0319
	DTLZ6	2	**0.2826**	**0.3739**	1.8949	1.0420	2.6166	0.7696	4.6779	1.2463
		3	**1.2833**	**0.2710**	2.9273	0.4893	1.2966	0.4552	3.0290	0.9259
		5	**0.7897**	**0.2869**	2.5206	0.6990	1.6732	0.6577	2.9527	1.1470
	DTLZ7	2	**0.0081**	**0.0113**	0.0444	0.0254	0.0260	0.0382	0.5942	0.1295
		3	**0.0500**	**0.0261**	0.1635	0.1030	0.0853	0.0443	0.6159	0.0980
		5	**0.2821**	**0.0235**	0.5763	0.2356	0.4916	0.3096	0.7254	0.0781
Optimal-Random	DTLZ2	2	0.4220	0.2079	**0.0053**	**0.0020**	0.0090	0.0029	0.1244	0.1827
		3	0.3152	0.2285	**0.0517**	**0.0101**	0.0554	0.0120	0.2088	0.1247
		5	0.1619	0.0604	0.1582	0.0143	**0.1404**	0.0253	0.2758	**0.0078**
	DTLZ4	2	0.8335	0.8480	**0.0194**	**0.0160**	0.0526	0.0351	0.5851	0.4683
		3	0.7853	0.1831	**0.2662**	**0.0738**	0.2966	0.0857	0.5575	0.1704
		5	0.5789	0.1020	**0.4319**	0.1062	0.4730	0.0904	0.6047	**0.0801**
	DTLZ5	2	0.7489	0.4255	**0.0086**	**0.0024**	0.0094	0.0032	0.2086	0.2516
		3	0.3323	0.3085	**0.0064**	0.0018	0.0076	**0.0017**	0.1010	0.0845
		5	0.1890	0.2090	**0.0049**	**0.0019**	0.0055	0.0021	0.0251	0.0232
	DTLZ6	2	**0.0064**	0.0031	0.0077	**0.0013**	0.0081	0.0019	0.0147	0.0019
		3	0.0556	0.0868	**0.0075**	**0.0021**	0.0085	0.0029	0.0198	0.0104
		5	0.0396	0.0986	**0.0069**	0.0014	0.0085	**0.0012**	0.0171	0.0078
	DTLZ7	2	**0.0005**	0.0004	0.0013	**0.0003**	0.0020	0.0007	0.0177	0.0033
		3	0.0397	0.0093	**0.0365**	**0.0043**	0.0388	0.0058	0.1012	0.0124
		5	0.1910	0.0179	**0.1855**	**0.0141**	0.1825	0.0220	0.3404	0.0367

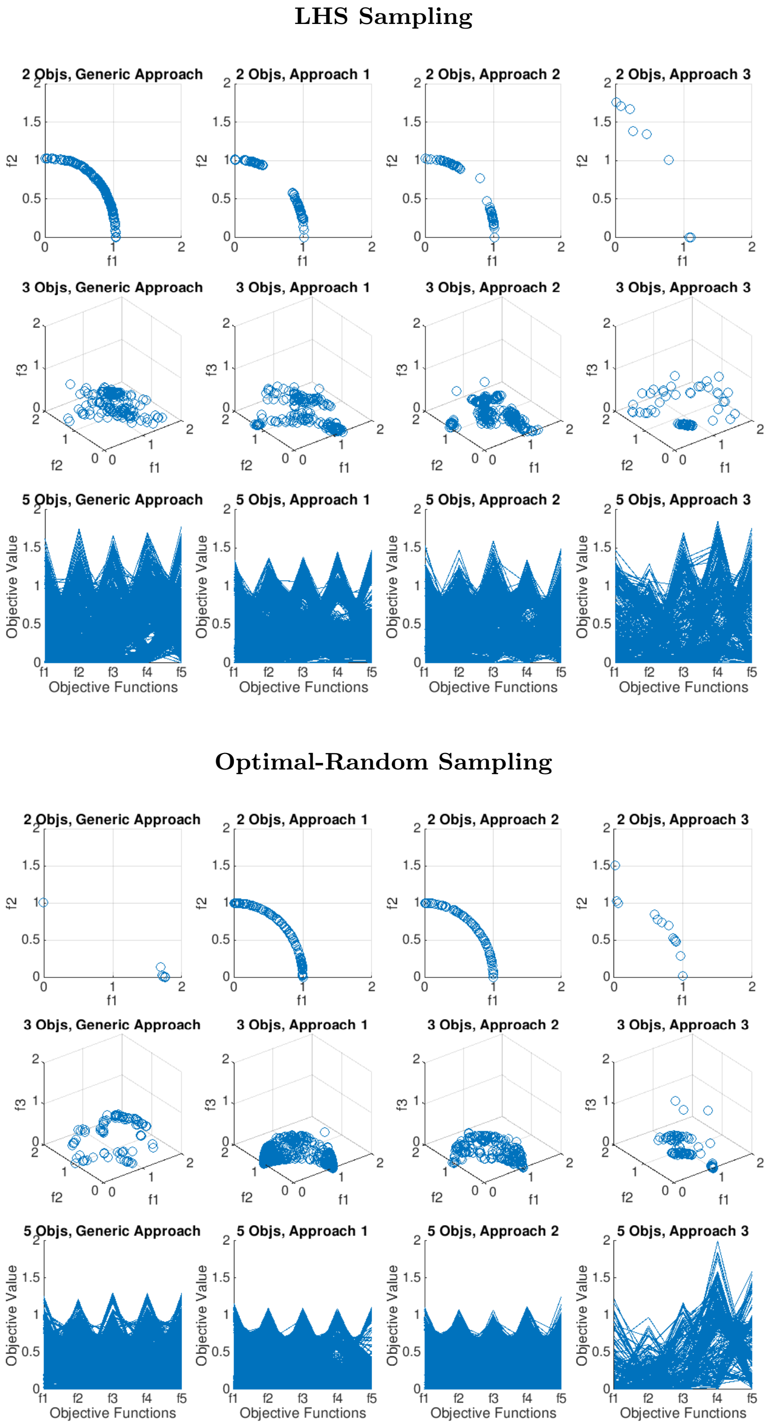

Fig. 4. Final solutions obtained of the run with the median IGD value using different approaches for LHS sampling (top three rows) and optimal-random sampling (bottom three rows) for the DTLZ2 problem.

5 Conclusions

We have considered offline data-driven optimization with evolutionary multiobjective optimization. We used Kriging to fit surrogate models to data and proposed and tested three approaches to utilize uncertainty information from Kriging models in the optimization. A comparison was done with several benchmark problems, sampling techniques and varying the number of objectives in solving offline data-driven multiobjective optimization problems. Adding uncertainty as one or more objectives showed improvements in the final solutions for certain problems in our benchmark testing. However, utilizing expected improvements as objectives (in Approach 3) did not seem to be effective in solving this kind of problems. The analysis also revealed that the solutions obtained in Approaches 1 and 2 are more accurate compared to the ones obtained using a generic approach (without uncertainty information).

Future work will include comparing the performance of the proposed approaches with bigger initial sample sizes, higher number of decision variables and higher number of objectives. Aiding the decision making process by giving a decision maker an option to select a final solution using the uncertainty information is another direction to work on. Moreover, filtering techniques can be applied to remove solutions with higher uncertainties. Testing on real-world data sets and exploring different ways to deal with uncertainties using other surrogate models will also be future research topics.

Acknowledgements

This research is related to the thematic research area Decision Analytics utilizing Causal Models and Multiobjective Optimization (DEMO) at the University of Jyvaskyla. This work was partially supported by the Natural Environment Research Council [NE/P017436/1].

References

1. Bezerra, L.C.T., López-Ibáñez, M., Stützle, T.: A large-scale experimental evaluation of high-performing multi- and many-objective evolutionary algorithms. Evolutionary Computation **26**(4), 621–656 (2018)
2. Blackwell, T., Branke, J.: Multiswarms, exclusion, and anti-convergence in dynamic environments. IEEE Transactions on Evolutionary Computation **10**(4), 459–472 (2006)
3. Castano, S., Antonellis, V.D.: Global viewing of heterogeneous data sources. IEEE Transactions on Knowledge and Data Engineering **13**(2), 277–297 (2001)
4. Chugh, T., Chakraborti, N., Sindhya, K., Jin, Y.: A data-driven surrogate-assisted evolutionary algorithm applied to a many-objective blast furnace optimization problem. Materials and Manufacturing Processes **32**(10), 1172–1178 (2017)
5. Chugh, T., Jin, Y., Miettinen, K., Hakanen, J., Sindhya, K.: A surrogate-assisted reference vector guided evolutionary algorithm for computationally expensive many-objective optimization. IEEE Transactions on Evolutionary Computation **22**(1), 129–142 (2018)

6. Chugh, T., Sindhya, K., Hakanen, J., Miettinen, K.: A survey on handling computationally expensive multiobjective optimization problems with evolutionary algorithms. Soft Computing (to appear, DOI: 10.1007/s00500-017-2965-0)
7. Coello, C., Lamont, G., Veldhuizen, D.: Evolutionary Algorithms for Solving Multi-Objective Problems. Springer, New York, 2nd edn. (2007)
8. Forrester, A., Sobester, A., Keane, A.: Engineering Design via Surrogate Modelling. John Wiley & Sons (2008)
9. Jeong, S., Obayashi, S.: Efficient global optimization (EGO) for multi-objective problem and data mining. In: 2005 IEEE Congress on Evolutionary Computation. vol. 3, pp. 2138–2145 Vol. 3 (2005)
10. Jin, Y.: Surrogate-assisted evolutionary computation: Recent advances and future challenges. Swarm and Evolutionary Computation **1**, 61–70 (2011)
11. Jin, Y., Wang, H., Chugh, T., Guo, D., Miettinen, K.: Data-driven evolutionary optimization: An overview and case studies. IEEE Transactions on Evolutionary Computation (to appear, DOI: 10.1109/TEVC.2018.2869001)
12. Jones, D.R., Schonlau, M., Welch, W.J.: Efficient global optimization of expensive black-box functions. Journal of Global Optimization **13**(4), 455–492 (1998)
13. Knowles, J.: ParEGO: A hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems. IEEE Transactions on Evolutionary Computation **10**(1), 50–66 (2006)
14. Li, K., Omidvar, M., Deb, K., Yao, X.: Variable interaction in multi-objective optimization problems. In: Handl, J., et al. (eds.) Parallel Problem Solving from Nature – PPSN XIV. pp. 399–409. Springer (2016)
15. Pan, S.J., Yang, Q.: A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering **22**(10), 1345–1359 (2010)
16. Pilat, M., Neruda, R.: Aggregate meta-models for evolutionary multiobjective and many-objective optimization. Neurocomputing **116**, 392–402 (2013)
17. Rasmussen, C., Williams, C.: Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning). The MIT Press (2005)
18. Regis, R.G.: Evolutionary programming for high-dimensional constrained expensive black-box optimization using radial basis functions. IEEE Transactions on Evolutionary Computation **18**(3), 326–347 (2014)
19. Sun, X., Gong, D., Jin, Y., Chen, S.: A new surrogate-assisted interactive genetic algorithm with weighted semisupervised learning. IEEE Transactions on Cybernetics **43**(2), 685–698 (2013)
20. Wang, H., Jin, Y., Jansen, J.O.: Data-driven surrogate-assisted multiobjective evolutionary optimization of a trauma system. IEEE Transactions on Evolutionary Computation **20**(6), 939–952 (2016)
21. Wang, H., Jin, Y., Sun, C., Doherty, J.: Offline data-driven evolutionary optimization using selective surrogate ensembles. IEEE Transactions on Evolutionary Computation (to appear, DOI: 10.1109/TEVC.2018.2834881)
22. Wang, H., Zhang, Q., Jiao, L., Yao, X.: Regularity model for noisy multiobjective optimization. IEEE Transactions on Cybernetics **46**(9), 1997–2009 (2016)
23. Wang, S., Minku, L.L., Yao, X.: Resampling-based ensemble methods for online class imbalance learning. IEEE Transactions on Knowledge and Data Engineering **27**(5), 1356–1368 (2015)
24. Zhang, Q., Liu, W., Tsang, E., Virginas, B.: Expensive multiobjective optimization by MOEA/D with Gaussian process model. IEEE Transactions on Evolutionary Computation **14**(3), 456–474 (2010)
25. Zitzler, E., Künzli, S.: Indicator-based selection in multiobjective search. In: Parallel Problem Solving from Nature - PPSN VIII. pp. 832–842. Springer (2004)