



Surrogate-Assisted NSGA-II Algorithm for Expensive Multiobjective Optimization

Mouadh Yagoubi, Hiba Bederina

► To cite this version:

Mouadh Yagoubi, Hiba Bederina. Surrogate-Assisted NSGA-II Algorithm for Expensive Multiobjective Optimization. Genetic and Evolutionary Computation Conference (GECCO 2023), Jul 2023, Lisbon, Portugal. 10.1145/3583133.3590746 . hal-04096875

HAL Id: hal-04096875

<https://hal.science/hal-04096875>

Submitted on 13 May 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Surrogate-Assisted NSGA-II Algorithm for Expensive Multiobjective Optimization

Mouadh Yagoubi

IRT SystemX, Palaiseau, France

mouadh.yagoubi@irt-systemx.fr

Hiba Bederina

hiba.bederina@gmail.com

ABSTRACT

Most real world problems are multiobjective by nature and expensive to evaluate. We propose in this work a surrogate assisted approach for multiobjective evolutionary algorithms by building a surrogate model on each objective. However, integrating a surrogate model within an optimization process generates complexity with additional hyper-parameters to tune. Empirical validation on standards MOO benchmark problems of a use case based on NSGA-II and surrogates using SVM regression shows a significant improvement of the optimization cost in terms of true objectives evaluations, especially for low budget. We also discuss the behavior of the proposed algorithm using different values of the parameter calibrating the use of the surrogate model.

KEYWORDS

Multiobjective optimization, Surrogate, Machine learning

ACM Reference Format:

Mouadh Yagoubi and Hiba Bederina. 2023. Surrogate-Assisted NSGA-II Algorithm for Expensive Multiobjective Optimization. In *Genetic and Evolutionary Computation Conference Companion (GECCO '23 Companion)*, July 15–19, 2023, Lisbon, Portugal. ACM, New York, NY, USA, 4 pages. <https://doi.org/10.1145/3583133.3590746>

1 INTRODUCTION

Many real world optimization problems are multi-objective by nature, involving the optimization of contradictory objectives (e.g., typically, cost and quality). In most cases, these objectives are not differentiable and noisy, making difficult the use of traditional optimization methods such as gradient descent. Hence, heuristics methods, like multi-objective evolutionary algorithms (MOEAs) [3] are good candidates for tackling such problems thanks to their robustness and efficiency in solving black-box optimization problems. Unfortunately, the main drawback of evolutionary algorithms (EAs) is their high cost in terms of number of function evaluation required to reach a satisfactory solution. This drawback can become prohibitive for those real world problems that generally require the use of heavy numerical simulations to compute a single value of the objective function and/or the constraints. To deal with this

issue, several works have focused in the last two decades on using surrogate modeling to reduce the global optimization cost, allowing the use of computationally expensive methods like evolutionary algorithms. In the literature, several machine learning techniques have been considered to build these surrogate models such as : quadratic models, Gaussian processes (or Kriging [7]) or support vector machine. The reader is referred to [5] for a comprehensive survey of surrogate assisted evolutionary optimisation. However, these works concern in their majority single objective optimization, the case of multi-objective optimization which suffers more from computational issues, remains less explored. In last years, the development of surrogate assisted multi-objective optimization have received a growing interest. (see [6] for more details on metamodels for multiobjective optimization). Several techniques for surrogate multiobjective optimization have been explored they can be categorized into two main strategies. The first one is based on optimizing a surrogate model built on the original problem, without any update during the optimization. The implementation of such strategy is straightforward, however the quality of the obtained solutions depends only on the quality of the surrogate model initially built on randomly generated solutions. The second strategy, commonly proceeds by replacing the objective function with the surrogate model, computing the true objectives on carefully selected points, and retraining the model from time to time using recently evaluated individuals. In [10] several approximation techniques (linear regression, SVM regression and neural networks) have been tested on two state-of-the art MOEAs namely NSGA-II and IBEA. in [9] a mono surrogate approach based on SVM classifiers was used within state-of-the-art MOEA to reduce the number of evaluation of objective function. Recently, a generalized probabilistic surrogate-assisted framework was proposed in [2] that covers both single and multi objective evolutionary optimization. However, the main challenge of surrogate-assisted optimization in general, and evolutionary multi-objective optimization in particular, is to find a good trade-off between the expected gain in terms of optimization global cost and the quality of obtained solutions. On the one hand, an excessive use of the surrogate model will certainly lead to a lower quality of obtained solutions than the original algorithm. On the other hand, reducing dramatically the call to the surrogate model will call into question the usefulness of its use within the optimization process.

At the same time, the performance assessment the surrogate-based optimization should not lose sight of the basic assumption behind using surrogate which is that the original objective functions is computationally expensive and hence the global number of functions evaluation is limited. Consequently, the assessment of surrogate contribution in improving the cost of the optimization should be considered at several stages of the optimization process, especially

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

GECCO '23 Companion, July 15–19, 2023, Lisbon, Portugal

© 2023 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-0120-7/23/07.

<https://doi.org/10.1145/3583133.3590746>

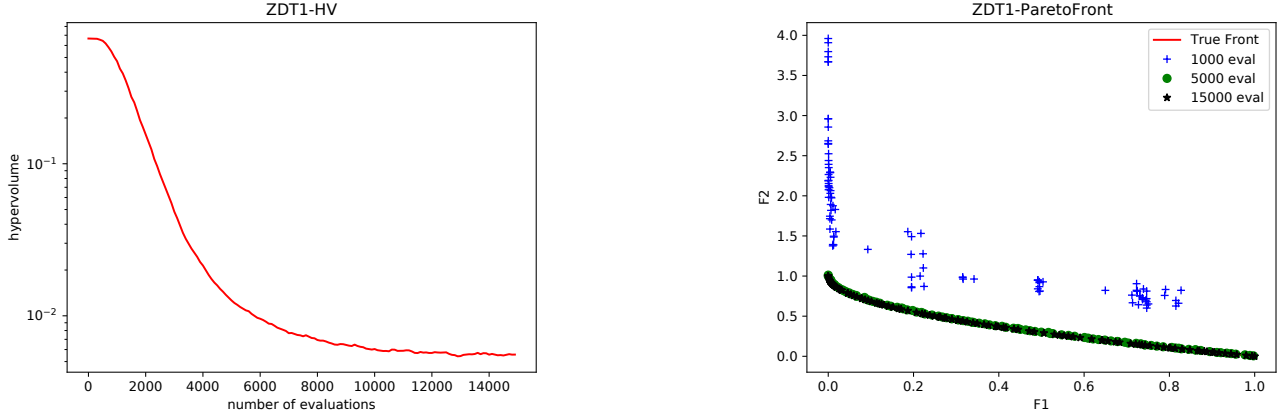


Figure 1: Left: Hypervolume evolution of the ZDT1 function. Right: successive fronts at different evaluations number.

at the first stages, to match with limited budget contexts. For example, for the ZDT1 function (a bi-objective function test with a convex Pareto front), the Pareto Front is almost reached after 5000 evaluations (see figure 1). While the hypervolume quality indicator will continue to improve after 5000 evaluations, this will not correspond to any discovery of a new region of the optimal Pareto set, but only to a better distribution of the points over the already reached front. This is yet another instance of the well known exploration/exploitation dilemma, specific to the multiobjective context. For those real world problems with expensive evaluation function, improving the distribution of solutions on an already reached Pareto-Front may not be of interest compared with discovering new regions toward optimal solutions.

The goal of this paper is to experimentally study the influence of using surrogate models within MOEAs to reduce the global optimization cost. A special focus will be made on the first stage of the optimization which may correspond better to expensive real world optimization problems with a limited evaluation budget. The next section will present briefly the proposed framework for surrogate assisted multiobjective optimization. Section 3 will discuss the obtained results. Finally, section 4 concludes this paper.

2 PROPOSED FRAMEWORK

We present in this section the general framework that we propose for surrogate-assisted MOEAs, which is summarized in algorithm 1. This framework can be instantiated using any MOEA and any machine learning technique for regression. To simplify the notation, the parent population size is set to be equal to the offspring population size and is equal to (N). The algorithm begins by initializing an initial population P_0 of size N , and evaluating it using the real (expensive) evaluation function. An Archive A is initialized by making a deep copy of P_0 , and a surrogate model based on a regression model is built on each objective m . For each generation t , N_{Iter} iterations are performed using the surrogate model: an offspring population Q^i is generated from the parent population P^i and evaluated based on the surrogate. Then, the replacement procedure will determine the next parent population P^{i+1} . The Q^{all} population

will store all generated points based on the surrogate evaluation during the N_{Iter} iterations. Then, a clustering approach is applied on the Q^{all} population (of size $N * N_{Iter}$) and N clusters are created. This procedure is detailed in algorithm 2, and was inspired by the work published in [2], in which the use of clustering has improved the preservation of diversity during the surrogate-based iterations. The K-means algorithm is used to identify the N clusters based the Q^{all} population. For each cluster, the dominance based ranking is performed to determine a winner per cluster that will join the offspring population Q_t . Doing so, we obtain the Q_t population of size N with surrogate influenced solutions, that we hope are more promising than those generated directly without using surrogate iterations. The Replacement procedure is then applied based on the real evaluation function to obtain the next parent population of the MOEA P_{t+1} . Before moving to the next generation, the archive A is updated with the newly evaluated solutions of Q_t and the surrogate is updated for each objective m . This procedure is repeated until the stopping criteria of the MOEA is met, typically a fixed number of evaluations of the real objective function.

3 EXPERIMENTS

3.1 Experimental Settings

The experiments were conducted on two bi-objectives test suites namely : ZDT [12] and LZ09 [8]. For all instances, the number of variables was set to 10. All reported results are averages (or aggregations) over 11 independent runs. All runs were stopped after 15000 evaluations with the real (expensive) objective function. We use NSGA-II [4] as a baseline MOEA algorithm, and Support Vector Regression (SVR) [11] as a surrogate approach. For SVR, we use RBF as a kernel function. The parent population size was set to $N = 100$ and is equal to the offspring population size. Several experiments were conducted with different values of the N_{Iter} parameter : 1, 3, 6 and 10. For the case with $N_{Iter} = 1$ we don't use the clustering approach at the end of the surrogate iterations, the offspring population generated based on the surrogate evaluation $Q^{i=1}$ will directly be used to infill the offspring population Q_t . The performance measurement considered for all experiments is the

Algorithm 1 Reg-Surrogate-MOEA

```

1:  $t \leftarrow 0, P_0 \leftarrow DoE(), |P_0| \leftarrow N$  (Initial population)
2:  $A \leftarrow \emptyset$  (Initialize Archive)
3: Real-Eval ( $P_0$ )
4:  $A \leftarrow A \cup P_0$  (Initialize Archive)
5:  $Surrogate_m.fit(A)$  for  $m \in 1, \dots, M$ 
6: for  $t=1 \dots N_{gen}$  do
7:    $P_t^{i=0} \leftarrow copy(P_t)$ 
8:   for  $i=0 \dots N_{Iter}$  do
9:      $Q^i \leftarrow variation(Select(P^i))$ 
10:    Surrogate-Eval( $Q^i$ )
11:     $P_t^{i+1} \leftarrow replacement(Q^i \cup P^i)$ 
12:     $Q^{all} \leftarrow Q^{all} \cup Q^i$ 
13:  end for
14:   $Q_t \leftarrow ClusteringBasedSelection(Q^{all}, size(Q_t))$ 
15:  Real-Eval( $Q_t$ )
16:   $P_{t+1} \leftarrow replacement(Q_t \cup P_t)$ 
17:   $A \leftarrow A \cup Q_t$ 
18:   $Surrogate_m.fit(A)$  for  $m \in 1, \dots, M$ 
19: end for

```

Algorithm 2 ClusteringBasedSelection(P, N)**Require:** Population P , $T = size(P)$, $N =$ number of clusters

```

1:  $S \leftarrow \emptyset$ 
2:  $Cluster_n \leftarrow Kmeans(P)$ .  $n = 1, \dots, N$ 
3: for  $n = 1 \dots N$  do
4:    $Winner_n \leftarrow dominanceRanking(Cluster_n)$ 
5:    $S = S \cup (Winner_n)$ 
6: end for
7: return  $S$ 

```

Hypervolume indicator [13], known to be consistent with Pareto dominance. Other performance indicators such as the epsilon indicator or IGD were considered in preliminary experiments, and they lead to the same conclusions. Consequently, they are not reported in this paper. For the baseline NSGA-II algorithm used in this series of experiments, we used the implementation of the JmetalPy framework [1] with a tournament size of 2.

3.2 Results

The final results obtained on ZDT and LZ09 by the variants of surrogate assisted NSGA-II instances are summarized in table 1. We report the average hypervolume indicator as well as the standard deviation at different stages of the optimization process: 1000, 5000 and 15000 evaluation respectively. We compare different values of the N_{Iter} parameter, which indicates the number of iterations (or generations) performed using the surrogate model as an evaluator instead of the real evaluation function. For each line in the table, the cells of best obtained results are colored gray. A first global observation that can be made from these results is that at 1000 evaluations the surrogate assisted versions significantly outperforms the original version for almost all instances, except ZDT2 and ZDT6 functions, for which the baseline NSGA-II and the surrogate-based variants are equivalent. This general observation confirms the contribution of the surrogate in accelerating the convergence process

Algo <i>N_{Iter}</i>	SVMregNSGA-II					NSGA-II	
	1	3	6	10	10		
ZDT1							
1000	5.09e-01 ± 7.62e-02	4.15e-01 ± 6.60e-02	4.28e-01 ± 8.43e-02	4.87e-01 ± 9.12e-02	5.05e-01 ± 6.36e-02		
5000	1.37e-02 ± 2.13e-03	1.81e-02 ± 2.77e-03	2.66e-02 ± 3.89e-03	4.46e-02 ± 1.08e-02	1.32e-02 ± 1.20e-03		
15000	5.58e-03 ± 1.67e-04	6.35e-03 ± 2.94e-04	7.92e-03 ± 8.81e-04	1.19e-02 ± 2.30e-03	5.56e-03 ± 2.78e-04		
ZDT2							
1000	3.33e-01 ± 5.55e-17	3.31e-01 ± 5.10e-03	3.33e-01 ± 5.55e-17	3.24e-01 ± 2.46e-02	3.33e-01 ± 5.55e-17		
5000	2.37e-02 ± 1.76e-02	1.29e-02 ± 3.92e-03	2.69e-02 ± 2.66e-02	1.53e-01 ± 1.50e-01	7.24e-02 ± 1.12e-01		
15000	3.33e-03 ± 1.55e-04	5.36e-03 ± 2.57e-04	5.59e-03 ± 4.00e-04	5.92e-03 ± 6.77e-04	5.28e-03 ± 2.08e-04		
ZDT3							
1000	6.52e-01 ± 1.17e-01	4.85e-01 ± 7.99e-02	4.90e-01 ± 9.15e-02	5.25e-01 ± 1.52e-01	6.43e-01 ± 6.13e-02		
5000	2.59e-02 ± 2.61e-02	2.36e-02 ± 1.48e-02	1.73e-02 ± 3.50e-03	2.32e-02 ± 9.17e-03	2.01e-02 ± 1.65e-02		
15000	7.86e-03 ± 1.55e-02	3.80e-03 ± 3.30e-04	3.96e-03 ± 3.79e-04	4.14e-03 ± 3.87e-04	2.81e-03 ± 1.16e-04		
ZDT6							
1000	3.26e-01 ± 0.00e+00	3.26e-01 ± 0.00e+00	3.26e-01 ± 0.00e+00	3.26e-01 ± 0.00e+00	3.26e-01 ± 0.00e+00		
5000	3.26e-01 ± 0.00e+00	3.26e-01 ± 0.00e+00	3.26e-01 ± 0.00e+00	3.26e-01 ± 0.00e+00	3.26e-01 ± 0.00e+00		
15000	4.68e-02 ± 1.21e-02	1.82e-01 ± 3.74e-02	2.02e-01 ± 4.41e-02	1.90e-01 ± 2.62e-02	4.95e-02 ± 1.04e-02		
LZ09-F1							
1000	1.68e-01 ± 3.01e-02	1.10e-01 ± 1.45e-02	9.56e-02 ± 1.08e-02	1.00e-01 ± 7.89e-03	1.71e-01 ± 1.72e-02		
5000	4.29e-02 ± 6.26e-03	5.02e-02 ± 6.69e-03	5.86e-02 ± 9.72e-03	6.61e-02 ± 8.15e-03	4.06e-02 ± 7.15e-03		
15000	1.85e-02 ± 2.74e-03	3.16e-02 ± 4.86e-03	4.23e-02 ± 9.58e-03	4.95e-02 ± 7.18e-03	1.79e-02 ± 1.03e-03		
LZ09-F2							
1000	6.38e-01 ± 2.47e-02	5.44e-01 ± 7.11e-02	5.17e-01 ± 7.32e-02	4.96e-01 ± 7.54e-02	6.48e-01 ± 1.77e-02		
5000	1.99e-01 ± 4.58e-02	1.90e-01 ± 5.42e-02	2.85e-01 ± 7.98e-02	2.99e-01 ± 1.08e-01	2.74e-01 ± 9.70e-02		
15000	1.57e-01 ± 3.96e-02	1.51e-01 ± 3.98e-02	2.06e-01 ± 7.31e-02	2.21e-01 ± 1.12e-01	1.85e-01 ± 5.51e-02		
LZ09-F3							
1000	4.13e-01 ± 2.57e-02	2.93e-01 ± 2.22e-02	2.26e-01 ± 2.13e-02	2.12e-01 ± 1.69e-02	4.24e-01 ± 4.38e-02		
5000	1.44e-01 ± 1.20e-02	1.32e-01 ± 8.07e-03	1.43e-01 ± 9.58e-03	1.61e-01 ± 1.87e-02	1.38e-01 ± 1.38e-02		
15000	6.68e-02 ± 6.79e-03	1.01e-01 ± 4.77e-03	1.15e-01 ± 7.82e-03	1.30e-01 ± 1.69e-02	9.62e-02 ± 1.33e-02		
LZ09-F4							
1000	4.11e-01 ± 3.73e-02	2.79e-01 ± 4.17e-02	2.11e-01 ± 3.33e-02	2.11e-01 ± 2.38e-02	3.78e-01 ± 2.21e-02		
5000	1.20e-01 ± 7.04e-03	1.14e-01 ± 1.30e-02	1.53e-01 ± 1.84e-02	1.44e-01 ± 1.33e-02	1.10e-01 ± 8.04e-03		
15000	7.23e-02 ± 3.04e-03	8.78e-02 ± 9.97e-03	9.88e-02 ± 7.91e-03	1.08e-01 ± 8.32e-03	7.23e-02 ± 3.57e-03		
LZ09-F5							
1000	4.06e-01 ± 3.01e-02	2.41e-01 ± 2.48e-02	1.91e-01 ± 9.68e-03	1.72e-01 ± 1.65e-02	4.01e-01 ± 2.93e-02		
5000	1.09e-01 ± 7.59e-03	1.06e-01 ± 6.92e-03	1.17e-01 ± 8.56e-03	1.22e-01 ± 1.24e-02	1.13e-01 ± 7.95e-03		
15000	7.21e-02 ± 9.59e-03	7.80e-02 ± 6.12e-03	8.62e-02 ± 5.21e-03	9.79e-02 ± 8.89e-03	6.72e-02 ± 2.92e-03		
LZ09-F7							
1000	6.66e-01 ± 1.11e-16	6.66e-01 ± 1.11e-16	6.62e-01 ± 1.11e-02	6.47e-01 ± 4.00e-02	6.66e-01 ± 1.11e-16		
5000	3.71e-01 ± 1.04e-01	3.96e-01 ± 8.10e-02	3.69e-01 ± 5.46e-02	4.65e-01 ± 9.38e-02	4.08e-01 ± 8.39e-02		
15000	2.84e-01 ± 5.59e-02	3.00e-01 ± 9.47e-02	2.51e-01 ± 1.72e-02	3.28e-01 ± 3.93e-02	2.53e-01 ± 6.65e-02		
LZ09-F9							
1000	6.66e-01 ± 1.11e-16	6.62e-01 ± 7.05e-03	6.54e-01 ± 1.70e-02	6.24e-01 ± 3.19e-02	6.66e-01 ± 1.11e-16		
5000	4.70e-01 ± 4.87e-02	4.49e-01 ± 6.84e-02	4.46e-01 ± 4.62e-02	4.37e-01 ± 3.04e-02	4.87e-01 ± 8.04e-02		
15000	2.64e-01 ± 4.04e-02	3.21e-01 ± 4.00e-02	3.13e-01 ± 4.39e-02	3.43e-01 ± 3.82e-02	2.63e-01 ± 3.30e-02		
LZ09-P9							
1000	3.32e-01 ± 2.17e-03	3.29e-01 ± 8.38e-03	3.23e-01 ± 1.73e-02	3.12e-01 ± 1.96e-02	3.32e-01 ± 1.64e-05		
5000	1.98e-01 ± 4.04e-02	2.28e-01 ± 3.31e-02	2.40e-01 ± 5.75e-02	2.27e-01 ± 4.42e-02	2.66e-01 ± 6.36e-02		
15000	1.54e-01 ± 3.09e-02	1.93e-01 ± 3.62e-02	1.98e-01 ± 5.97e-02	1.86e-01 ± 7.40e-02	2.21e-01 ± 3.93e-02 height		

Table 1: Mean and Standard Deviation of hypervolume indicator over 11 trials after 1000, 5000 and 15000 real evaluations.

toward optimal solutions. After 5000, the baseline NSGA-II seems to catch up the surrogate based variants for some instances (ZDT1, LZ09-F1 and LZ09-F4) and at 15000 evaluations, NSGA-II and at least one of the surrogate variants seems to reach the same Pareto front, except for LZ09-F9 for which NSGA-II is outperformed by the surrogate variants. Regarding the N_{Iter} parameter, we can observe that we can not draw a clear recommendation about its best value. Depending on the instance, the value of N_{Iter} that gives the best result may differ. We can even clearly observe that a simple change in N_{Iter} parameter can change the ranking between the baseline NSGA-II algorithm and the surrogate-based approach. Moreover, another observation can be made regarding the hypervolume indicator and its evolution over the optimization process: for some instances, the best choice regarding the N_{Iter} parameter may be different according to the number of real evaluations reached. For example for LZ09-F1 and LZ09-F5 functions (see figure 2), at the first stage of the optimization process (before 2000 evaluation), setting N_{Iter} at a high value (6 or 10) seems to be better than lower values (1 or 3). However, as the optimization proceeds, and after 2500 evaluations, lower values of N_{Iter} allow to get most promising solutions and the variant with $N_{Iter}=1$ is competitive with the baseline NSGA-II at 15000 evaluations.

3.3 Discussion

The use of performance indicators like Hypervolume, IGD or Epsilon indicator is very important to compare MOEAs with more than one objective. However, the simple comparison of these indicators after a fixed number of function evaluations, often very high, might not be appropriate to compare surrogate-based approaches.

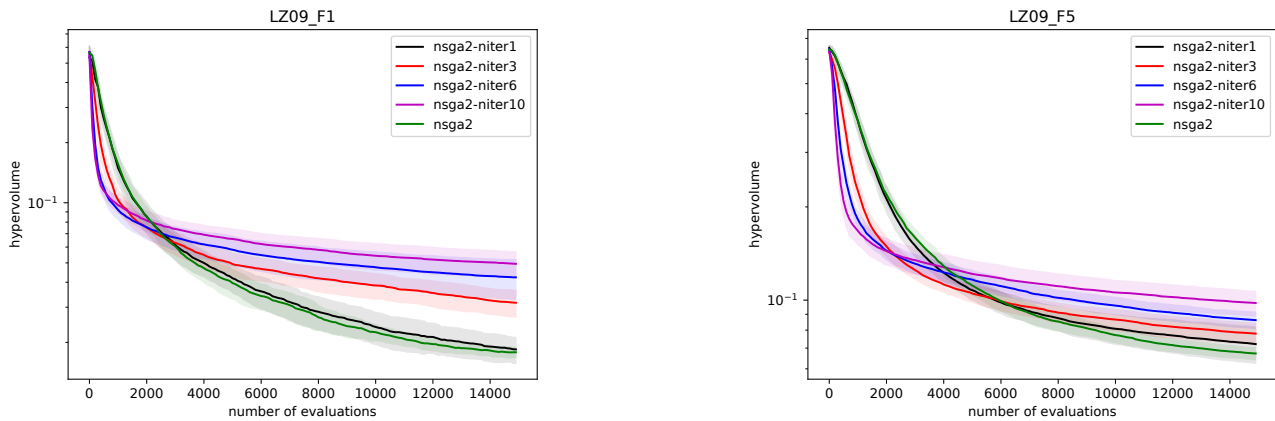


Figure 2: Mean Hypervolume (lines) and standard deviation (shaded regions) for surrogate assisted variants and baseline NSGA-II on LZ09-F1 and LZ09-F5 functions. Plots of other instances are available in Appendix.

Observing the evolution of the performance indicators over the evolutionary process allows to better understand the impact of the choices made on the quality of obtained solutions. From the results reported in the previous section, it is clear that the parameters calibrating the use of the surrogate have to be set dynamically, as the best value of the N_{Iter} parameter change over the optimization process. Also, we cannot predict beforehand the behaviour of the algorithm when the surrogate parameters are set dynamically. This opens on the challenging question of designing an efficient self-adaptive approach to make the best choices and avoid an off-line arbitrary setting. On the other hand, as discussed in section 1, the interpretation of the performance indicator like hypervolume, should be considered with a certain precaution. After a certain number of evaluations, the improvement of the hypervolume indicator will not necessarily reflect a discovery of new promising points toward the Pareto front. Hence, for real world problems with expensive evaluation functions, spending a considerable computational time (two thirds of the total evaluation budget for ZDT1 as illustrated in figure 1) in finding a good distribution of an already reached Pareto front may not be of great interest, even using a good fined-tuned surrogate-based approach within the MOEA.

4 CONCLUSION

This paper has empirically studied the behaviour of surrogate assisted MOEAs. Investigations have been conducted using the well known state-of-the-art algorithm NSGA-II, on standard MOO benchmarks namely : ZDT and LZ09. The proposed approach builds a surrogate model for each objective using the SVM regression approach. Experiments were conducted using different hyper-parameter settings to calibrate the use of the surrogate. The results demonstrate a significant acceleration when using the surrogate model, especially at relatively low budgets in terms of evaluation number (1000 evaluations). However, the setting of parameters calibrating the use of the surrogate model remains a challenging issue, and reported results show that no conclusive recommendation can be

made, as the best values change from an instance to another. On-going and future work is to propose an on-line tuning approach to calibrate the surrogate model. The design of such approach will require to extend the comparative study by considering other MOEAs, and by comparing with other surrogate techniques, such as neural network, and to other surrogate approaches, such as building a single surrogate model for all objectives at once. Another perspective is to extend all of the above to benchmarks with three and more objective functions, such as the DTLZ benchmark.

REFERENCES

- [1] Antonio Benitez-Hidalgo, Nebro, et al. 2019. jMetalPy: A Python framework for multi-objective optimization with metaheuristics. *Swarm and Evolutionary Computation* 51 (2019), 100598.
- [2] Julian Blank and Kalyanmoy Deb. 2022. GPSAF: A Generalized Probabilistic Surrogate-Assisted Framework for Constrained Single-and Multi-objective Optimization. *arXiv preprint arXiv:2204.04054* (2022).
- [3] Kalyanmoy Deb. 2011. *Multi-objective optimisation using evolutionary algorithms: an introduction*. Springer.
- [4] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation* 6, 2 (2002), 182–197.
- [5] Yaochu Jin. 2005. A comprehensive survey of fitness approximation in evolutionary computation. *Soft computing* 9, 1 (2005), 3–12.
- [6] Joshua Knowles and Hirotaka Nakayama. 2008. Meta-modeling in multiobjective optimization. *Multiobjective optimization: Interactive and evolutionary approaches* (2008), 245–284.
- [7] Daniel G Krige. 1951. A statistical approach to some basic mine valuation problems on the Witwatersrand. *Journal of the Southern African Institute of Mining and Metallurgy* 52, 6 (1951), 119–139.
- [8] Hui Li and Qingfu Zhang. 2008. Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II. *IEEE transactions on evolutionary computation* 13, 2 (2008), 284–302.
- [9] Ilya Loshchilov, Marc Schoenauer, and Michèle Sebag. 2010. A mono surrogate for multiobjective optimization. In *Proc. GECCO*. 471–478.
- [10] Martin Pilát and Roman Neruda. 2011. ASM-MOMA: Multiobjective memetic algorithm with aggregate surrogate model. In *2011 IEEE Congress of Evolutionary Computation (CEC)*. IEEE, 1202–1208.
- [11] Alex J Smola and Bernhard Schölkopf. 2004. A tutorial on support vector regression. *Statistics and computing* 14 (2004), 199–222.
- [12] Eckart Zitzler, Kalyanmoy Deb, and Lothar Thiele. 2000. Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary computation* 8, 2 (2000), 173–195.
- [13] Eckart Zitzler and Lothar Thiele. 1999. Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. *IEEE transactions on Evolutionary Computation* 3, 4 (1999), 257–271.