A Case Study on Multi-Criteria Optimization of an Event Detection Software under Limited Budgets[*]

Martin Zaefferer[1], Thomas Bartz-Beielstein[1], Boris Naujoks[1], Tobias Wagner[2], and Michael Emmerich[3]

[1] Faculty for Computer and Engineering Sciences
Cologne University of Applied Sciences, 51643 Gummersbach, Germany
firstname.lastname@fh-koeln.de

[2] Institute of Machining Technology (ISF)
TU Dortmund University, 44227 Dortmund, Germany
wagner@isf.de

[3] Leiden Institute for Advanced Computer Science
Leiden University, The Netherlands,
emmerich@liacs.nl

Abstract. Several methods were developed to solve cost-extensive multi-criteria optimization problems by reducing the number of function evaluations by means of surrogate optimization. In this study, we apply different multi-criteria surrogate optimization methods to improve (tune) an event-detection software for water-quality monitoring. For tuning two important parameters of this software, four state-of-the-art methods are compared: S-Metric-Selection Efficient Global Optimization (SMS-EGO), S-Metric-Expected Improvement for Efficient Global Optimization SExI-EGO, Euclidean Distance based Expected Improvement Euclid-EI (here referred to as MEI-SPOT due to its implementation in the Sequential Parameter Optimization Toolbox SPOT) and a multi-criteria approach based on SPO (MSPOT).

Analyzing the performance of the different methods provides insight into the working-mechanisms of cutting-edge multi-criteria solvers. As one of the approaches, namely MSPOT, does not consider the prediction variance of the surrogate model, it is of interest whether this can lead to premature convergence on the practical tuning problem. Furthermore, all four approaches will be compared to a simple SMS-EMOA to validate that the use of surrogate models is justified on this problem.

1 Introduction

The time required for a process feedback can play a crucial role in many fields of industrial optimization. Complex and expensive real-world processes or time consuming simulations lead to large evaluation times. This restricts optimization processes to only a very limited number of such evaluations. Moreover,

[*] The original publication is available at www.springerlink.com.

almost all industrial optimization tasks feature more than one quality criterion. Techniques from multi-criteria decision making, evolutionary multi-criteria optimization (EMO) in particular, were developed during the last decade to solve such tasks. The necessity to combine EMO techniques and optimization methods such as EGO [13] or SPO [1], which require a very small number of function evaluations only, should be self-evident. The application of such methods to real-world problems in industrial optimization provides a reasonable way to assess their feasibility. In contrast to artificial test functions, it allows for an assessment of the practical relevance for these kinds of problems.

In this paper we focus on four different tuning methods which are applied to tune an anomaly detection software for water quality management. This problem is usually handled by receiver operator characteristic (ROC) analysis. Due to specific limitations of the software concerned, this can not be applied in the classical way. Rather, the ROC curve should be approximated by Multi-Criteria Optimization (MCO) methods. That means, the ROC curve can be interpreted as a Pareto front. Interpreting ROC curves from the multi-criteria optimization perspective is an established approach in computational intelligence, see, e.g., [17].

In Sec. 2, we will summarize the former work performed in relevant research fields. The specific problem is presented in Sec. 3. The tuning algorithms (based on different SPO and EGO implementations) are described in Sec. 4. Section 5 describes the experimental setup, whereas the analysis is presented in in Sec. 6. Finally, Sec. 7 gives a summary of findings and an outlook on future work.

2 Former research

Surrogate modeling is not a new topic in optimization. Jin [12] provides a comprehensive overview of single-objective optimization with surrogate models. While methods like EGO or SPO for single criteria optimization are well established, the application of surrogate modeling procedures for multiple objectives is more recent.

2.1 Surrogate modeling in multi-criteria optimization

In MCO, several approaches employ surrogate modeling. One example is the well established ParEGO by Knowles [15]. An overview of surrogate modeling in MCO is given by Knowles and Nakayama [16]. To balance exploration and exploitation in case of a limited budget, several methods employ infill criteria based on expected improvement (EI). Two things are required for defining such a criterion: the definition of the improvement and an algorithm to compute its expectation [22]. Since negative improvements are not possible, dominated solutions should yield an improvement of zero. As large variances potentially result in large improvements and large deteriorations are not penalized, these criteria also focus on the exploration of uncovered areas of the search space. It is of interest to see if this kind of additional exploration is desirable for the problem at hand. In particular, it remains to be seen whether there is already sufficient exploration done due to the initial design or due to the requirement of covering a whole set of Pareto optimal points.

2.2 ROC analysis

This work deals with tuning the event detection software CANARY [10, 19][4] which tries to detect anomalies in water quality data. The core algorithm in CANARY compares the difference between a predicted value and the most recently measured value to a user defined threshold. If the threshold is exceeded, an alarm is triggered. ROC provides means to select a threshold of a classifier based on trade-off between its True Positive Rate (TPR) and False Positive Rate (FPR). In the case of CANARY, TPR is the hit rate which is based on the number of correctly recognized events. FPR on the other hand is the false alarm rate. False alarms occur whenever the algorithm detects an event when actually none exists.

The ROC curve shows the trade-off between TPR and FPR. Usually, it is drawn based on the threshold value of the classifier. This means, depending on the chosen threshold value one receives different pairs of TPR/FPR values which can be connected to a curve. To evaluate the performance of a classifier, the Area Under Curve (AUC) can be used. The worst possible classifier will have an AUC of 0.5, since all pairs of TPR and FPR will be on the straight line between the two extreme points of the curve. This performance would be equal to random guessing. The best possible classifier will have an AUC of 1, which means there is a configuration where no false alarms occur, all events are identified (cf. Fig. 1).

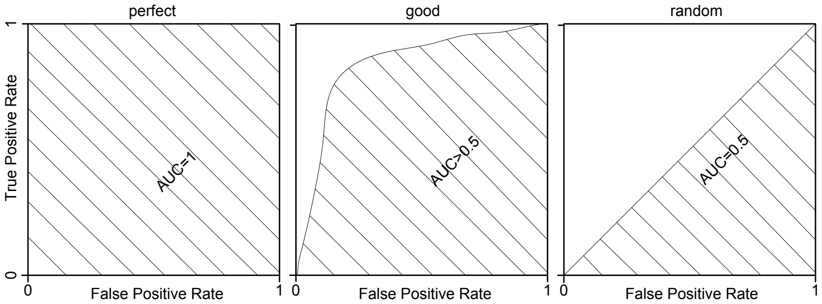

Fig. 1: *ROC curves for classifiers of different quality. Leftmost is the case of perfect classification, the rightmost is random guessing.*

In the case of CANARY, this form of measuring the performance can not be used, since the threshold value used in CANARY cannot be chosen independently. Therefore, each different setting of the threshold has to be considered as

[4] For documentation, manuals and source code of CANARY see: `https://software.sandia.gov/trac/canary`

a new classifier. The ROC curve can then be used to compare performance of the different classifiers. Consequently, the threshold value is one of the parameters to be optimized.

2.3 MCO in ROC Analysis

The ROC curve can be interpreted as a Pareto front, although it would classically only represent the Pareto front of an MCO problem with one dimensional decision space (i.e. the decision threshold being the only decision variable). However, it is reasonable to apply MCO methods for other cases, for instance when different classifiers are to be compared, or the threshold is not independent of the classification process. This is the case in the problem described in this paper.

Applying MCO for ROC analysis is not a new topic. Kupinski and Anastasio [17] considered performances of the solutions returned from a multi-criteria objective genetic optimization as series of optimal (sensitivity, specificity) pairs, which can be thought of as operating points on a ROC curve. Fieldsend and Everson [7] used MCO to construct the ROC curve for a vector of parameters (including the threshold) of a binary classifier, thus analyzing several different classifiers. In a second study they discuss the application of MCO for the ROC analysis of a multi class problem [5]. A survey of MCO in ROC analysis can be found in the work of Everson and Fieldsend [6].

3 Problem Description

The problem to be solved in this paper is the tuning of a software designed for anomaly detection in water quality management: CANARY. It was developed to detect anomalies (or events) in water quality time series data. It implements several different algorithms for time series prediction, pattern matching, and outlier detection. The main concept is to employ a time series algorithm to predict the next time step, and afterwards to distinguish whether the real value deteriorates from the predicted value sufficiently to declare it an outlier or anomaly.

We will tune the two relevant parameters window size and threshold value. The window size defines how many values are used for the prediction, while the threshold value defines how much deviation between measured and predicted value are sufficient to declare an outlier. Both parameters have previously been tuned in different ways. Firstly, they have been tuned by a step-by-step procedure [19] which unfortunately does not consider interactions between parameters. Secondly, another study [23] tuned them with model based optimization, considering interactions, but only used a single criteria approach, which basically combined the objectives False Alarm Rate and Hit Rate to a weighted sum.

Usually, as described by Murray et al. [19], a classical ROC analysis would be performed. The AUC would be used as a single quality criterion. This approach is not perfectly viable in this case, as the threshold value is not independent of the prediction process. Therefore, it is a more reasonable approach to add the threshold to the list of tuned parameters and apply multi-criteria optimization.

For this reason, we will mainly use MCO-terminology in the following (e.g., Pareto front instead of ROC curve).

4 Algorithm Description

Four different tuning algorithms are in the focus of this study. Due to the similarity to the AUC, the hypervolume is applied as a criterion in all but one of these approaches. Two of them are based on R-code (SPOT package), two are MATLAB implementations (SMS-EGO and SExI-EGO). All four share the following basic workflow:

1. Evaluate an initial design of n points on the target problem (CANARY)
2. Build models (here: Kriging) for each objective
3. Use models to determine the next design point to be evaluated, based on a certain infill criterion
4. Evaluate design point and update non dominated set
5. Iterate 2-4

The four tuning algorithms differ in the type of the invoked infill criterion. Three algorithms use different multi-criteria EI concepts. The fourth is a straightforward approach that, instead of aggregating the objective values from the models, tries to optimize these separately with common MCO methods.

4.1 MEI-SPOT

This multi-criteria expected improvement approach is the only approach that does not use hypervolume as a criterion. The implementation is based on MATLAB code of Forrester et al. [9]. MEI-SPOT is based on the integration over the non-dominated area and an Euclidean distance to the next point on the front. While Forrester et al. use a dominating variant (e.g. improvement considers only points that dominate existing Pareto-optimal solutions), the implementation used here uses an augmenting variant (i.e. improvement is also reported when a point is added to the front, without dominating an existing Pareto-optimal solution). The different formulations for this distinction are detailed by Keane [14]. This approach is time consuming due to the integration. It can also have issues with the scaling of different objectives, since it is based on the Euclidean distance.

4.2 SExI-EGO

The S-Metric Expected Improvement [3] computes the expected increment in hypervolume for a point, given a non-dominated set. Its exact computation is described in [4]. It is differentiable, rewards high variances [4], and is continuous over the whole search domain. A disadvantage is the high effort of its exact computation, in particular when more than two objectives are considered.

4.3 SMS-EGO

SMS-EGO, as suggested by Ponweiser et al. [20], employs a hypervolume based infill criterion as well. Thereby, a potential solution is computed using the lower confidence bound $\hat{y}_{pot} = \hat{y} - \alpha\hat{s}$, where $\hat{y}$ is the mean value predicted by the Kriging model, $\hat{s}$ is the variance, α is a gain factor for the variance. This approach may also explore unvisited regions of the design space, but without requiring the tedious integration of the previous approaches. It thus scales better with increasing objective dimension.

If the resulting $\hat{y}_{pot}$ is ϵ-dominated or dominated, SMS-EGO will assign a penalty value. If it is non-dominated, the hypervolume contribution will be used. This approach avoids plateaus of the criterion, but integrates non differentiable parts. For more details see Ponweiser et al. [20] and Wagner et al. [22].

4.4 MSPOT

MSPOT is a multi-criteria approach based on the Sequential Parameter Optimization Toolbox SPOT (cf. Zaefferer et al. [24]). It does not employ any form of expected improvement, or other forms of using the variance for exploration. The surrogate models of the different objectives are exploited by using a multi-criteria optimization algorithm (for instance: SMS-EMOA or NSGA-II).

This will yield a population of promising points. One or more points of these are chosen for evaluations on the real target function. This selection is based on non-dominated sorting and the individual hypervolume contribution. As the original approach [24] could lead to clustering of solutions in the objective space, the available points have to be considered when calculating the hypervolume contributions. For this purpose, the known points are reevaluated on the surrogate model.

In contrast to the other approaches in the study, this one does not promote exploration as much, since the variance measure computed by the Kriging model will not be used. On the other hand, the approach is not limited to surrogate modeling methods that yield a variance for each candidate. Of course, the variance can easily be added to MSPOT, as well as be removed from SMS-EGO ($\alpha = 0$) or the integration-based algorithms ($\hat{s} = 0$).

The optimization process of MSPOT is not a completely new idea. Especially, two similar approaches suggested previously have to be mentioned. Firstly, Voutchkov and Keane [21] employed NSGA-II to generate promising solutions in a quite similar optimization loop. In contrast to MSPOT, they used Euclidean distance to ensure evenly spaced points on the front. Instead of considering distance to known points, they suggest a larger number points in each loop, which also ensures a wider spread on the final front. The second similar approach is presented by Jeong and Obayashi [11]. While they also optimize the objectives separately, they employ the single objective EI criterion for each objective, thus optimizing a vector of EI values.

4.5 SMS-EMOA

In addition to the four approaches above, a simple SMS-EMOA will be considered (cf. Beume et al. [2]). The results from this optimizer are used as a baseline for the comparison. In general, surrogate optimization methods are expected to outperform a non-surrogate SMS-EMOA, particularly on small budgets.

5 Experimental Setup

The following research questions are to be treated for the CANARY problem in this study.

1. Can multi-criteria methods produce a front of parameter settings that help an operator to choose parameters for the CANARY event detection software?
2. Which kind of tuner is recommendable?
3. What aspects of a tuner affect its performance?
4. Is the use of surrogate models advantageous?
5. Can previous findings about the tuners be confirmed?
6. How are Pareto optimal solutions spread in the design space?

To answer these questions, several experiments were conducted. Their setup is described in the following.

5.1 Time Series Data

Two different sets of raw data are used. The first set is used to train CANARY (i.e. to tune the parameters), the second is used for validation of the resulting settings on unseen data. Additionally, from each of those sets, 3 different instances are generated, where each contains simulated (i.e. superimposed) events to be detected by CANARY. The data sets considered are available within the CANARY software package.

Training Data The data recorded over a first month at a specific measurement station is used as training data. Four different sensor values are used (pH-Value, Conductivity, Total Organic Carbon, Chlorine). The time interval between measurements is five minutes. This results in about 9 000 time steps for each of the four sensors.

As the data-set contains no events known beforehand (which is a typical problem for any available real-world data), events have to be simulated and incorporated in the time series. Therefore, 3 data sets are created from the raw data, each containing superimposed square waves (with smoothed transition) of different event strengths: 0.5, 1, and 1.5. These strengths indicate the amplitude of the events, and are multiplied to the standard deviation of the original signal. Figure 2 presents raw data and data with events for two sensor value as an example.

As can be seen from the left part of Fig. 2, the raw data (i.e. without events) is rather strongly affected by background changes. In general, these background

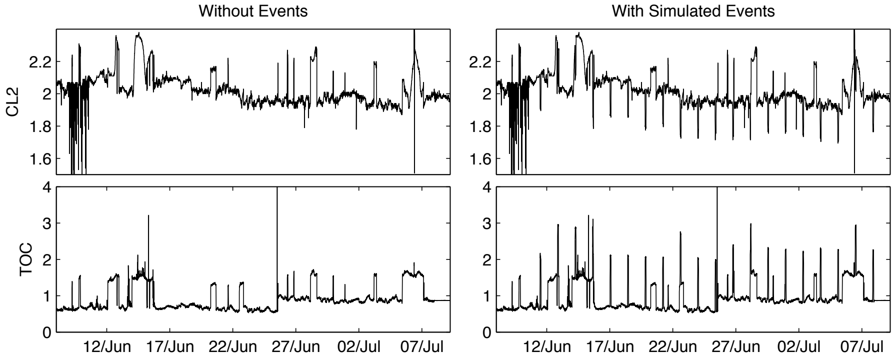

Fig. 2: *Example of time series data as used in the experiments. Raw data without events (left) and with superimposed events of strength 1.5(right). CL2 is Chlorine, TOC the Total Organic Carbone.*

changes are irregularly distributed over time and always switch back and forth for each of the signals. Obviously, such changes make event detection extremely difficult.

Validation Data The validation data is similar to the training data, as it is the second month of data from the same measurement station. As could be expected, it provides a very similar background behavior with some sudden jumps. These jumps, however, are more numerous than in the training data, which is expected to lead to higher false alarm rates on the validation data.

5.2 Optimization Problem Configuration

As mentioned earlier, three different data sets are considered, each with a different event strength. Additionally, CANARY is tuned in 3 different configurations, where each configuration uses a different time series prediction algorithm. These are: Time Series Increment TSI, Linear Prediction Correction Filter LPCF and Multi-Variate Nearest Neighbor MVNN. For more details on these algorithms, which are implemented in CANARY, see the corresponding documentation [19] and the manual [10]. Therefore, $3 \times 3 = 9$ instances are to be optimized. The optimization problem is multi-objective, where both decision and solution space are two dimensional: The window size and the threshold are tuned, to yield a minimal FPR and a maximal TPR value. Since all tuning methods in this study do minimization, TPR is negated. The problem is not noisy, as the algorithms employed in the event detection software are deterministic.

There are two nice features of the problem, which avoid issues of algorithm configuration.

1. The choice of the reference point. With this problem the worst case is known: Zero for TPR and one for FPR. To avoid extreme points overlapping with

the reference point, the latter was chosen to be [0.1,1.1] since TPR ranges from -1 to 0 due to the negation.

2. Scaling of objective space is not an issue here, as both objectives have the same range. They only differ in that way, that TPR is maximized and FPR minimized. Scaling has not to be considered in the algorithms.

5.3 Tuning Methods Configuration

All algorithms are configured to use approximately the same settings, i.e.:

- initial design size: 21
- number of points added in each step: 1
- number of maximal evaluations of the target function: 80
- surrogate model: `DACE`-Kriging [18]

The optimization method to find the best point (with or without expected improvement) differs. Some criteria aggregate the different objectives into a single-objective infill criterion, which is then optimized. Therefore, SExI-EGO, SMS-EGO and MEI-SPOT invoke a local optimization method, restarting in several partitions of the design space. In contrast to this, MSPOT uses SMS-EMOA to optimize the surrogate models of the objectives without aggregating their information.

While both the `R` and the `MATLAB` implementations use `DACE`-Kriging [18] there are small differences in the implementations. This includes differences between the inbuilt local optimization methods (e.g. simplex, gradient based) used during model building and optimization.

The SMS-EMOA employed as a baseline comparison is configured to also use a starting population of 21 points. All other settings are left at defaults.

6 Analysis

The results of the experiments are depicted in Fig. 3. It shows the resulting hypervolume of each tuner for each problem instance. The hypervolume values are recalculated with respect to the reference point $[0, 1]$ to have the ranges comparable to the AUC values. Plots with the original reference point used during tuning look alike and do not show major differences. As can be seen, there are no significant differences between the performance of SMS-EGO, MSPOT, and SExI-EGO. In comparison, MEI-SPOT and SMS-EMOA perform worse. For the SMS-EMOA, this was expected and can be blamed to not invoking a surrogate model. The Euclidean EI criterion employed in MEI-SPOT, on the other hand, was already reported to be less viable due a non-monotonicity with the dominance relation [22].

It can be observed that the event strength has an improving influence on the detection performance of the Pareto optimal solutions. This is expected, as stronger events should be easier to identify. The same can be observed for the algorithm MVNN, which provides best overall detection results. Both observations are in line with earlier reported behavior in the work on tuning CANARY single objectively [23]. An optimal performance would be leading to a hypervolume of exactly one. Realistically, this is not obtainable. The gap between the best front's hypervolume and the theoretical optimum is largely due to the fact that the FPR rate is strongly affected by the sudden jumps in the time series data. Besides this, the results are in a similar range of TPR and FPR values as found in the earlier mentioned single-objective tuning of CANARY.

The results discussed above have been received on the training data set only. To validate our findings, we invoked the validation data set as well. All points of the final Pareto front were re-evaluated on the validation data set. It can be observed that performance differences become smaller while variances increase. The former particularly holds for incorporating MVNN in CANARY. In general, it can be noticed that the received hypervolume decreases on the validation data. This can mainly be blamed on a slightly different background behavior of the validation data set, as described in Sec. 5.1. While there is a strong performance drop for nearly all instances, the results for MVNN and an event strength of 1.5 only decrease slightly. A similar behavior was observed for single-objective tuning of CANARY in an earlier work by Zaefferer [23]. Still, the validation data shows the same relations as the training data, considering the performance of different tuners. In the briefness of this paper we focus on the training data results, since differences can mainly be blamed on different background behavior in the two time frames (e.g. more sudden jumps in the second month, compared to the first month).

Table 1 provides the average number of points on a single Pareto front. Note, that 30 to 50 percent of the points on a front are actually dominated, if being re-evaluated on the validation data.

Table 1: *Average number of points on a Pareto front. The second line shows how many of those points remain, after being reevaluated on validation data.*

	MEI-SPOT	MSPOT	SMS-EGO	SExI-EGO	SMS-EMOA
Training Data	38.70	29.27	23.11	22.92	21.00
Validation Data	22.91	18.13	14.54	14.27	12.54

As mentioned above, MSPOT, SMS-EGO, and SExI-EGO do not show significant differences in their results. One main distinction between these approaches is that MSPOT does not make use of the variance produced by the `DACE` model, and therefore lacks exploration. It might therefore be the case that MSPOT performs as good as the other methods, because the initial design already provides enough exploration of the design space so that it is sufficient to spent all sequential evaluations on purely exploiting the surrogate models prediction.

To test this, two additional experiments are performed. Firstly, the MSPOT experiment is repeated with a much smaller initial design of just 5 points (labeled MSPOTSMALL), to validate whether a smaller initial design can deteriorate results. Secondly, the SMS-EGO experiment was repeated disregarding any variance information. To this end, the gain α is set to zero. Therefore, instead of

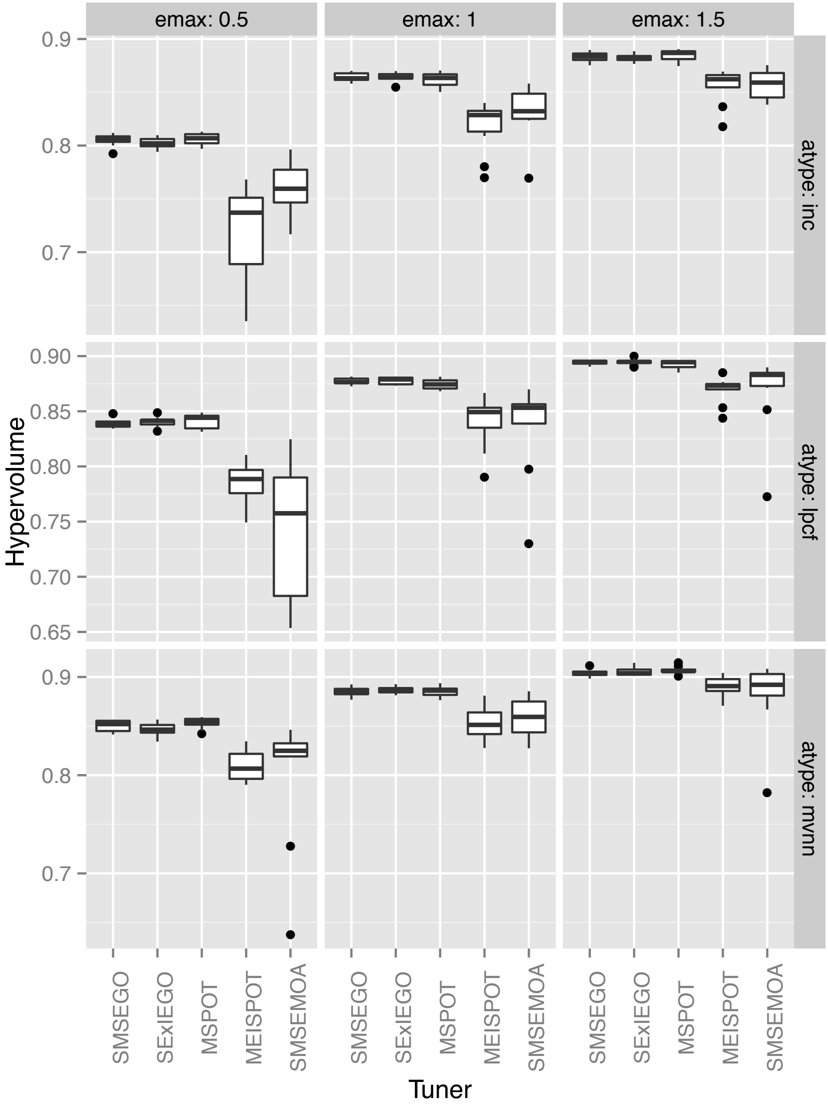

Fig. 3: *Boxplot of first results on training data. The hypervolume is computed with a reference point of zero for TPR and one for FPR. Larger hypervolumes are better. emax is the event strength, atype is the algorithm type used in CANARY.*

the lower confidence bound $\hat{y}_{pot} = \hat{y} - \alpha\hat{s}$ the potential solution will be $\hat{y}_{pot} = \hat{y}$. This approach will be labeled as SMS-EGOg0. The resulting hypervolumes on training data are depicted in Fig. 4. The smaller initial design in fact decreases performance of MSPOT, however, the margin is quite small. Furthermore, a comparable performance of SMS-EGO with or without taking the variance in consideration is observed.

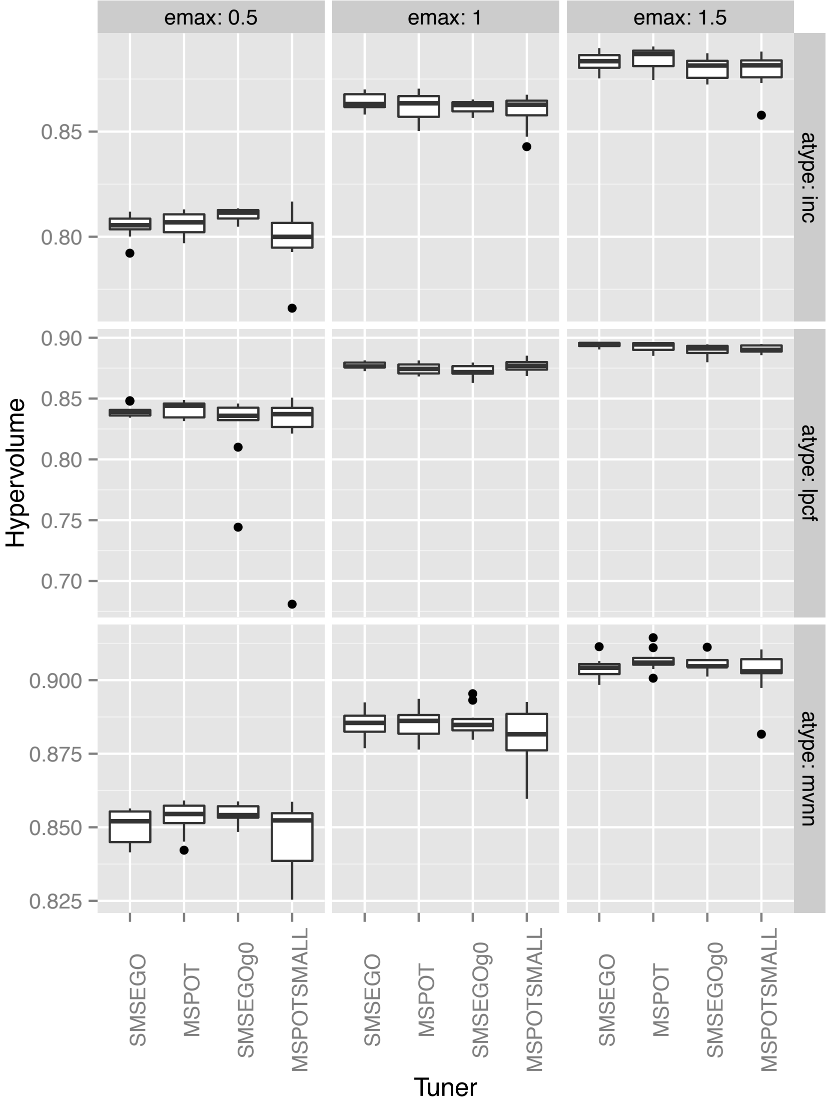

Fig. 4: *Boxplot of additional results on training data, comparing results of follow-up experiments (i.e. MSPOTSMALL and SMSEGOg0). The hypervolume is computed with a reference point of zero for TPR and one for FPR. Larger hypervolumes are better. emax is the event strength, atype is the algorithm type used in CANARY.*

In some instances SMS-EGO even performs better without incorporating the variance information. There seems to be no strong need for the additional exploration in this case. Such observations are normally expected for unimodal problems, while more exploration should be profitable on multi modal problems. It might further be considered that additional exploration is already inherent in the selection process as not one single optimum, but a set of points is demanded.

To visualize the problem landscape, Fig. 5 shows contour plots of reference DACE-models for each objective. These models were built by combining the designs of all algorithms and selecting some representatives based on the distance to an optimized Latin hypercube design. Whereas, the models seem to have a rather unimodal shape, there are clusters of optimal solutions due to a slightly oscillating behavior in the plateau regions. This effect can be observed using the model predictions and the actual data. As a consequence, the approximation of the knee region with window sizes between 200 and 400 and a threshold between 1.0 and 1.5 should be easy, whereas the extreme ones might become a multimodal problem.

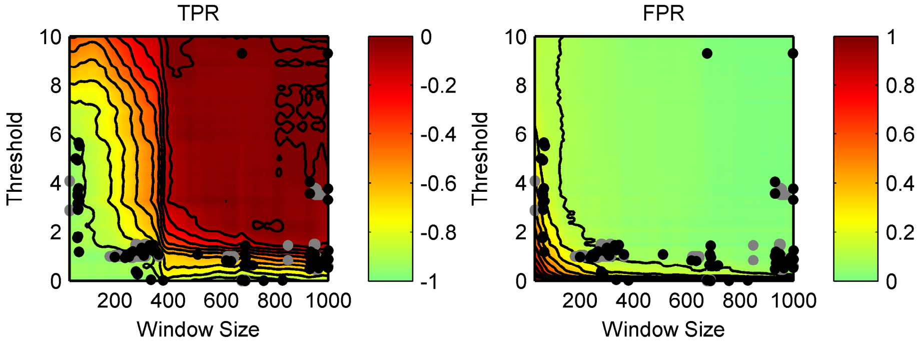

Fig. 5: Problem landscape of both objectives. Contours show DACE-model based on representatives from all evaluations on this instance (Algorithm MVNN and event strength 1.5). Black dots show all real pareto optimal solutions found. Grey dots show Pareto optimal solutions on the model, yielded with grid sampling.

7 Summary and Outlook

In this study, we tested different approaches based on surrogate optimization to tune an event detection software. Most of the analysis was focused on the results of training data, since the results on validation data mainly provided similar results as on the training data. The surrogate optimization approaches are mostly able to outperform a baseline SMS-EMOA. The MEI-SPOT approach proved to be the exception from this observation, which confirms earlier findings by Wagner et al. [22]. This approach of calculating the expected improvement for multiple criteria seems to be unfavorable.

There was no decisive difference between the other tested approaches, regardless whether variance was used in the approach (SMS-EGO and SExI-EGO) or not (MSPOT and SMS-EGO with zero gain). Plots of the model structures seem to indicate an almost unimodal fitness landscape for both objectives. This indicates that the additional exploration by variance might not be needed here, since the fitness landscape is easy to approximate without additional exploration of the design space. Since it was neither a disadvantage, it might be interesting to test the lower confidence bound in MSPOT for future experiments.

This study showed that the problem of tuning CANARY can reasonably be solved by multi-criteria methods. The produced results yield reasonable FPR and TPR values, which are comparable to previous results achieved by single-objective optimization. Here, however, the approximation of a Pareto front offers more flexibility for the operator in charge.

It has to be noted that only points on the convex hull of the ROC curve can be considered to be optimal in some sense. Any point below that hull might be considered to be improvable [8]. Future work should investigate if concavities in the ROC curve can be repaired for the application described here.

The concentration on certain regions of a Pareto front might be a topic for future research as well. An operator might be more interested in the knee region of the Pareto front, and less on extreme values, which might cause intolerable numbers of false alarms. Focusing on a subset of the Pareto front might also reduce the required budget.

Acknowledgments This work has been kindly supported by the Federal Ministry of Education and Research (BMBF) under the grants MCIOP (FKZ 17N0311) and CIMO (FKZ 17002X11). In addition, the paper is based on investigations of project D5 “Synthesis and multi-objective model-based optimization of process chains for manufacturing parts with functionally graded properties” as part of the collaborative research center SFB/TR TRR 30, kindly supported by the Deutsche Forschungsgemeinschaft (DFG).

References

1. T. Bartz-Beielstein, K. E. Parsopoulos, and M. N. Vrahatis. Design and analysis of optimization algorithms using computational statistics. *Applied Numerical Analysis and Computational Mathematics (ANACM)*, 1(2):413–433, 2004.
2. N. Beume, B. Naujoks, and M. Emmerich. SMS-EMOA: Multiobjective selection based on dominated hypervolume. *European Journal of Operational Research*, 181(3):1653–1669, 2007.
3. M. Emmerich. *Single- and Multi-objective Evolutionary Design Optimization: Assisted by Gaussian Random Field Metamodels.* PhD thesis, Universität Dortmund, Germany, 2005.
4. M. Emmerich, A. Deutz, and J. Klinkenberg. Hypervolume-based expected improvement: Monotonicity properties and exact computation. In *Evolutionary Computation (CEC), 2011 IEEE Congress on*, pages 2147–2154. IEEE, 2011.
5. R. M. Everson and J. E. Fieldsend. Multi-class ROC analysis from a multi-objective optimisation perspective. *Pattern Recognition Letters*, 27(8):918 – 927, 2006.
6. R. M. Everson and J. E. Fieldsend. Multi-objective optimisation for receiver operating characteristic analysis. In Y. Jin, editor, *Multi-Objective Machine Learning*, volume 16 of *Studies in Computational Intelligence*, pages 533–556. Springer Berlin / Heidelberg, 2006.

7. J. E. Fieldsend and R. M. Everson. ROC Optimisation of Safety Related Systems. In J. Hernández-Orallo, C. Ferri, N. Lachiche, and P. A. Flach, editors, *ROCAI*, pages 37–44, 2004.
8. P. A. Flach and S. Wu. Repairing concavities in ROC curves. In *Proceedings of the 19th international joint conference on Artificial intelligence*, IJCAI'05, pages 702–707, San Francisco, CA, USA, 2005. Morgan Kaufmann Publishers Inc.
9. A. Forrester, A. Sobester, and A. Keane. *Engineering Design via Surrogate Modelling*. Wiley, 2008.
10. D. B. Hart, K. A. Klise, E. D. Vugrin, S. A. McKenna, and M. P. Wilson. Canary user's manual and software upgrades. Technical Report EPA/600/R-08/040A, U.S. Environmental Protection Agency, Washington, DC, 2009.
11. S. Jeong and S. Obayashi. Efficient global optimization (EGO) for multi-objective problem and data mining. In D. Corne et al., editors, *IEEE Congress on Evolutionary Computation*, pages 2138–2145. IEEE, 2005.
12. Y. Jin. A comprehensive survey of fitness approximation in evolutionary computation. *Soft Computing*, 9(1):3–12, 2005.
13. D. Jones, M. Schonlau, and W. Welch. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13:455–492, 1998.
14. A. Keane. Statistical improvement criteria for use in multiobjective design optimisation. *AIAA Journal*, 44(4):879–891, 2006.
15. J. Knowles. Parego: A hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems. *IEEE Transactions on Evolutionary Computation*, 10(1):50–66, January 2006.
16. J. D. Knowles and H. Nakayama. Meta-modeling in multiobjective optimization. In *Multiobjective Optimization*, pages 245–284. Springer, 2008.
17. M. A. Kupinski and M. A. Anastasio. Multiobjective genetic optimization of diagnostic classifiers with implications for generating receiver operating characteristic curves. *IEEE Transactions on Medical Imaging*, 18:675–685, 1999.
18. S. Lophaven, H. Nielsen, and J. Søndergaard. DACE—A Matlab Kriging Toolbox. Technical Report IMM-REP-2002-12, Informatics and Mathematical Modelling, Technical University of Denmark, Copenhagen, Denmark, 2002.
19. R. Murray, T. Haxton, S. A. McKenna, D. B. Hart, K. Klise, M. Koch, E. D. Vugrin, S. Martin, M. Wilson, V. Cruz, and L. Cutler. Water quality event detection systems for drinking water contamination warning systems—development, testing, and application of CANARY. Technical Report EPA/600/R-10/036, National Homeland Security Research Center, May 2010.
20. W. Ponweiser, T. Wagner, D. Biermann, and M. Vincze. Multiobjective optimization on a limited budget of evaluations using model-assisted -metric selection. In *PPSN*, pages 784–794, 2008.
21. I. Voutchkov and A. Keane. Multiobjective optimization using surrogates. In *Adaptive Computing in Design and Manufacture ACDM*, pages 167–175, 2006.
22. T. Wagner, M. Emmerich, A. Deutz, and W. Ponweiser. On expected-improvement criteria for model-based multi-objective optimization. *Parallel Problem Solving from Nature–PPSN XI*, pages 718–727, 2010.
23. M. Zaefferer. Optimization and empirical analysis of an event detection software for water quality monitoring. Master's thesis, Cologne University of Applied Sciences, May 2012.
24. M. Zaefferer, T. Bartz-Beielstein, M. Friese, B. Naujoks, and O. Flasch. Multi-criteria optimization for hard problems under limited budgets. In T. Soule et al., editors, *GECCO 2012 Proceedings*, pages 1451–1452, Philadelphia, Pennsylvania, USA, July 2012. ACM.