

Optimization of Costly Technical Applications with Surrogate-Based Optimization Algorithms

Bernd Luderer¹, Martin Stöcker^{1*}

¹ Technische Universität Chemnitz, Chemnitz, Germany

Abstract

We consider the solution of technical optimization problems involving highly time-consuming calculations. Usually, these possibly multiobjective optimization problems come along with almost no knowledge about the behaviour of the objective functions and many local optima. In most cases, the problems are not sufficiently solvable by traditional gradient-based optimization methods. Instead, surrogate-based heuristics provide a bundle of effective techniques to find the global optimum of black-box functions. The benefit of these algorithms is the fast convergence towards the global optimum at a minimal number of function evaluations, whereas a drawback originates from the significant overhead produced by the algorithm itself. In this paper, we present EMMOA, a surrogate-based optimization method for multiobjective optimization of black-box functions. The iteration process starts with some points generated by a Latin hypercube approach. Repeatedly, a model is trained to fit the existing data and exploited to find new points which are likely to yield better results. Several models are used for multiobjective optimization. The way new points are chosen from the models is of crucial importance for the convergence of the algorithm: the exploitation of the model and the investigation of unexplored regions of the feasible set have to be balanced. We examine the performance of EMMOA at a real-world problem.

Keywords: surrogate-based optimization, kriging, nonlinear multiobjective optimization, black-box optimization

*Contact: Dipl.-Math. Martin Stöcker, TU-Chemnitz, Reichenhainer Straße 41, D-09107 Chemnitz, Germany, E-Mail: martin.stoecker@mathematik.tu-chemnitz.de

1 Introduction

We would like to solve multiobjective constraint optimization problems which emerge from technical settings. A function $F(\mathbf{x}) : \mathbb{R}^N \mapsto \mathbb{R}^M$ has to be minimized with respect to some (possibly nonlinear) constraints. If M is greater than one the solution of this optimization problem is a discrete approximation of the set of optimal trade-offs called the *Pareto set*. This set can help to identify relationships among the objectives and represents the basis for the choice of an optimal design.

In a technical framework a function evaluation $F(\vec{x})$ corresponds to the calculation of a complex simulation process (e. g. the solution of a large set of differential equations). Thus the time to compute values of the objective functions for a given $\vec{x} \in \mathbb{R}^N$ may be considerable (e. g. the time to calculate chain forces is about half an hour in our example in Section 5) and so the number of points at which function F can be evaluated is limited (a reasonable number is 300 evaluations).

Naturally, there are many methods to solve such problems; popular methods are genetic algorithms which work very well with nonlinear constraints and more than one objective. Other optimization strategies transfer the problem to an single-criteria problem (e. g. by weighted sum approach) and use single-objective optimization to find a solution.

The mentioned algorithms have something in common: the time the optimization algorithm needs itself is negligible. But if the evaluation of the objective function is time-consuming the optimization algorithm can need some time itself. This leads us to the family of surrogate-based optimization algorithms.

We use a stochastic model to construct M surrogate functions corresponding to the M objective functions. These models are used in the next step to find new points which are likely to yield better objective values. This iteration process is continued until the algorithm converges – or, in practice, if the time given to the optimization is over.

The model which is used to describe the objective function is of particular importance. Since the objective functions are considered as black-box, meaning there is no knowledge about their behaviour, it has to be able to capture linear and nonlinear tendencies.

2 Kriging model

The Kriging approximation [Jones et al. \(1998\)](#) is a popular method used as a surrogate model for optimization. It is able to model real valued functions $f : \mathbb{R}^N \mapsto \mathbb{R}$. Let us assume we have a set of variables $\{\vec{x}^1, \vec{x}^2, \dots, \vec{x}^k\}$, $\vec{x}^i \in \mathbb{R}^N$, $i = 1, \dots, k$ and corresponding objective values $\vec{y} = (y^1, y^2, \dots, y^k)^\tau$, $y^i = f(\vec{x}^i) \in \mathbb{R}$.

We construct a model as a surrogate for f which is fast to compute. It describes the function value at an unknown point $\hat{\vec{x}}$ of the feasible set as a combination of the known function values: $\hat{y} = \hat{f}(\hat{\vec{x}}) = \sum_{i=1}^k \lambda^i y^i$.

For that the correlation between the given points $\vec{x}^i$ has to be measured: $\text{corr}(\vec{x}^i, \vec{x}^j) = \exp(-\sum_{n=1}^N \theta_n |\vec{x}_n^i - \vec{x}_n^j|^{p_n})$. The parameters θ_n , p_n ($\theta, \mathbf{p} \in \mathbb{R}^N$) stand for the range of influence of each point and the smoothness of the model in dimension n respectively. These parameters need to be tuned to the available data by maximizing the following maximum likelihood estimate:

$$L(\mu, \sigma^2, \theta, \mathbf{p}) = \frac{1}{(2\pi\sigma^2)^{\frac{k}{2}} \sqrt{\det(\mathbf{C})}} \exp \left[-\frac{(\vec{y} - \vec{1}\mu)^\tau \mathbf{C}^{-1} (\vec{y} - \vec{1}\mu)}{2\sigma^2} \right].$$

We use the following constraints for the parameters: $0 < \theta_n$, $0 < p_n \leq 2$. Matrix $\mathbf{C}$ contains the correlation of all points ($\mathbf{C} = \text{corr}(\vec{x}^i, \vec{x}^j)_{i,j=1}^k$) and $\vec{1} = (1, 1, \dots, 1)^\tau$. The parameters μ and σ can be approximated by a maximum-likelihood-estimate: $\hat{\mu} = \frac{\vec{1}^\tau \mathbf{C}^{-1} \vec{y}}{\vec{1}^\tau \mathbf{C}^{-1} \vec{1}}$, $\hat{\sigma} = \frac{(\vec{y} - \vec{1}\hat{\mu})^\tau \mathbf{C}^{-1} (\vec{y} - \vec{1}\hat{\mu})}{k}$.

If the model is trained an estimation of the function value at an untried point $\hat{\vec{x}}$ of the feasible set is obtained by

$$\hat{f}(\hat{\vec{x}}) = \hat{\mu} + \mathbf{r} \mathbf{C}^{-1} (\vec{y} - \vec{1}\hat{\mu}).$$

Vector $\mathbf{r} \in \mathbb{R}^k$ stands for the correlation of $\hat{\vec{x}}$ with all existing points: $r_i = \text{corr}(\vec{x}^i, \hat{\vec{x}})$. The reliability of the estimation $\hat{f}(\hat{\vec{x}})$ is evaluated by

$$\hat{s}^2(\hat{\vec{x}}) = \hat{\sigma}^2 \left[1 - \mathbf{r}^\tau \mathbf{C}^{-1} \mathbf{r} + \frac{(1 - \vec{1}^\tau \mathbf{C}^{-1} \mathbf{r})^2}{\vec{1}^\tau \mathbf{C}^{-1} \vec{1}} \right].$$

We have $\hat{s}^2(\hat{\vec{x}}) = 0$ if $\hat{\vec{x}}$ is already contained in the set of data points. So, a value of $\hat{s}^2(\hat{\vec{x}})$ close to 0 implies an accurate model in the neighbourhood of $\hat{\vec{x}}$. In contrast, if $\hat{s}^2(\hat{\vec{x}})$ tends to the value of $\hat{\sigma}^2$ the estimation might be bad.

3 Surrogate-Based Global Optimization

For a one dimensional optimization problem ($M = 1$) the optimization process starts with the choice of some arbitrary points of the feasible set. The corresponding objective values are computed and a Kriging model $\hat{f}$ as a surrogate of the objective is constructed. Subsequently, one or more new points $\vec{x}^i$ are sampled from the surrogate model and $F_1(\vec{x}^i)$ is calculated. The sampled points have to meet two requirements:

1. At first, the surrogate model maps only strong tendencies of the objectives. Thus, the model has to be improved at locations where it is inaccurate (*exploration*).
2. Secondly, we need to find regions of the feasible set that produce optimal values of the objectives. This task is called the *exploitation* of the model.

To solve 1 and 2 we employ the Weighted Expected Improvement criterion (WEI) which assesses the potential of every point in the feasible set:

$$\text{WEI}(\vec{x}) = w (\underline{f} - \hat{f}(\vec{x})) \Phi \left(\frac{\underline{f} - \hat{f}(\vec{x})}{\hat{s}(\vec{x})} \right) + (1 - w) \hat{s}(\vec{x}) \phi \left(\frac{\underline{f} - \hat{f}(\vec{x})}{\hat{s}(\vec{x})} \right).$$

Here $\underline{f}$ stands for the best objective value that occurred so far, $\phi(\vec{x})$ and $\Phi(\vec{x})$ denote the standard normal density and distribution function respectively. The estimation of the objective value is represented by $\hat{f}(\vec{x})$, whereas $\hat{s}(\vec{x})$ measures the reliability of the estimation.

The criterion balances the exploration and the exploitation of the surrogate model. Parameter $w \in [0, 1]$ weighs the probability for an improvement by point $\vec{x}$ and the reliability of the model in the neighbourhood of $\vec{x}$. In practice we advise to start with a value around 0 for w (that is to sample points which improve the model) and to increase w in each iteration, meaning to prefer points which are likely to yield better function values.

After selecting points from the model it can be updated by the new data and the next iteration cycle is started. Finally, we stop the iteration process if there is no further improvement of the objective values.

4 Enhanced Multiobjective Metamodel Optimization Algorithm

In Knowles (2006) J. Knowles proposes the use of surrogate-based algorithms for multiobjective optimization problems. A single surrogate model is constructed from the weighted sum of the values of all objectives. The parameters of the weighted sum approach are alternated to produce an approximation of the Pareto set. We extend the *ParEGO* algorithm of J. Knowles by the use of several surrogate models which turned out to be beneficial for optimization problems with a high number of objectives as well as for highly time-consuming applications.

Following Knowles (2006) we choose $11N - 1$ evenly distributed points from the feasible set by a Latin Hypercube method. A Kriging model is constructed from the initial point set. It is trained by maximizing the likelihood function presented in Section 2. We apply a genetic algorithm, but one can use any optimizer for nonlinear optimization problems. To avoid numerical problems and to accelerate the training process only a subset of the data points is used to train the surrogate models: in practice a number of 100 data points is enough. Since the model should be accurate in the neighbourhood of the Pareto set we choose training points which are close to that set. An appropriate selection criterion is the ranking-crowding scheme used in the multiobjective genetic algorithm NSGA-II (see Deb et al. (2000)).

In a first phase called *depth-first search* we look for the point with the highest value of WEI for each model separately:

$$\max_{\vec{x}} \text{WEI}_m(\vec{x}), \quad m = 1, 2, \dots, M.$$

Afterwards, the objectives of M points are computed in each step of the depth-first search. This produces a high model accuracy in the neighbourhood of the Pareto front of the optimization problem. In addition, it allows a parallelized computation of some function values.

In the second stage, the *breadth-first search*, points with high values of Weighted Expected Improvement are searched over all models, thus solving a multiobjective optimization problem:

$$\max_{\vec{x}} (\text{WEI}_1, \text{WEI}_2, \dots, \text{WEI}_M)^\tau.$$

This procedure supports the convergence towards the true Pareto front. To select points for sampling from the solution of the optimization problem a criterion for an even approximation of the Pareto set is used.

5 Optimization of a Chain Drive

We apply the optimization algorithm EMMOA to an optimization of a chain drive in a combustion engine. A chain drive powers the cam shaft and ancillary components. On the one hand we need to tighten the chain to prevent the loosening of the chain. Therefore a hydraulic tightener and two tensioner blades are enforced. On the other hand the chain has to be as loose as possible (to decrease the chain forces in all parts that have no contact to a structural element). Excitations from the crankshaft and the camshafts cause dynamic stresses. To decrease these forces we use two noncircular gears.

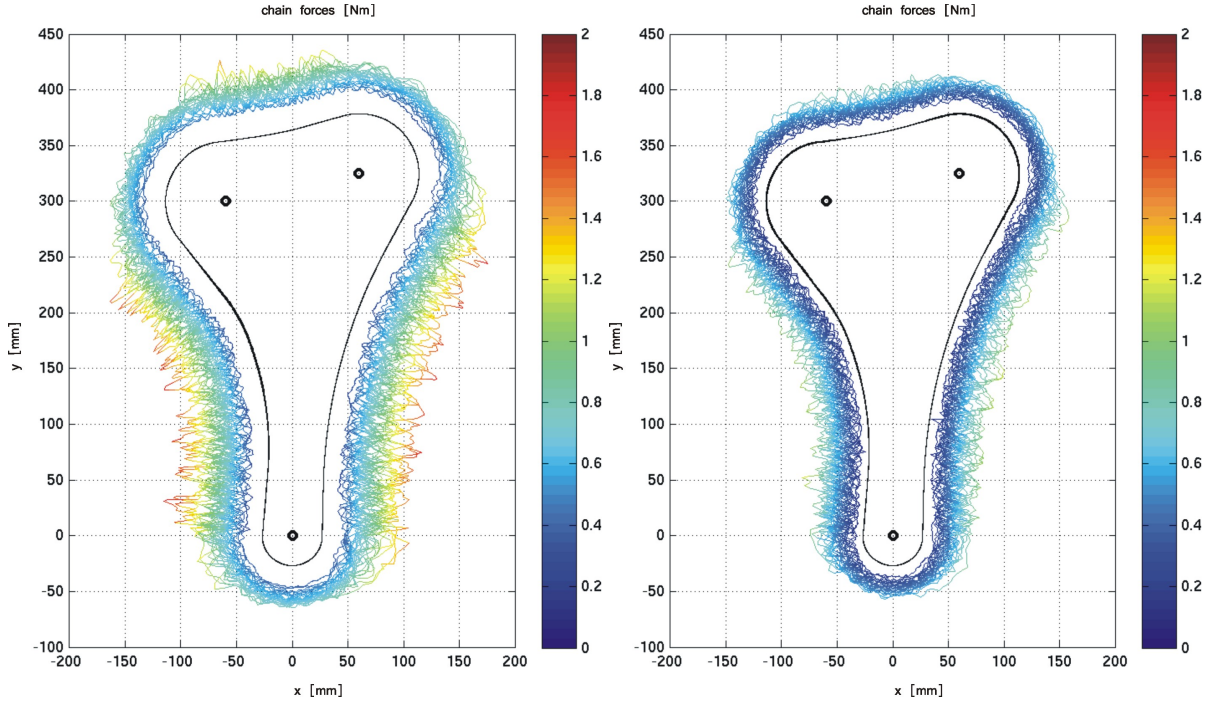


Figure 1: chain forces for several speeds before and after optimization

We intend to minimize the dynamic stress within the chain. This yields the following three objectives for the optimization:

- the minimization of the maximal chain forces over several speeds,
- the minimization of the difference of torsion between inlet and outtake camshaft and
- the minimization of the difference of the amplitudes of inlet and outtake camshaft.

We consider the use of two identical noncircular gears. The last objective ensures that some of the trade-offs in the Pareto set generate two gears with the same amplitude. The following variables with interval constraints are used for optimization:

- the gap size of the tightener,
- the tightener force,
- the amplitudes and
- the phase angles of the noncircular gears.

In addition, each variant has to satisfy the following restriction: the minimal chain force for each speed must not drop below a defined threshold, which guarantees that the chain does not become too loose.

The results obtained in comparison to other multiobjective optimization algorithms are very promising. In particular the low number of evaluations of the objective functions to reach the Pareto front should be mentioned. In Figure 1 the chain forces for several speeds are plotted on the chaindrive. The forces before optimization show high peaks in the area of the tighteners. After the optimization the forces are generally lower and more homogenous. Figure 2 depicts the distribution of the chain forces as a function of the crankshaft angle and the speed of the engine. Obviously, the amount of extremal chain forces is reduced considerably.

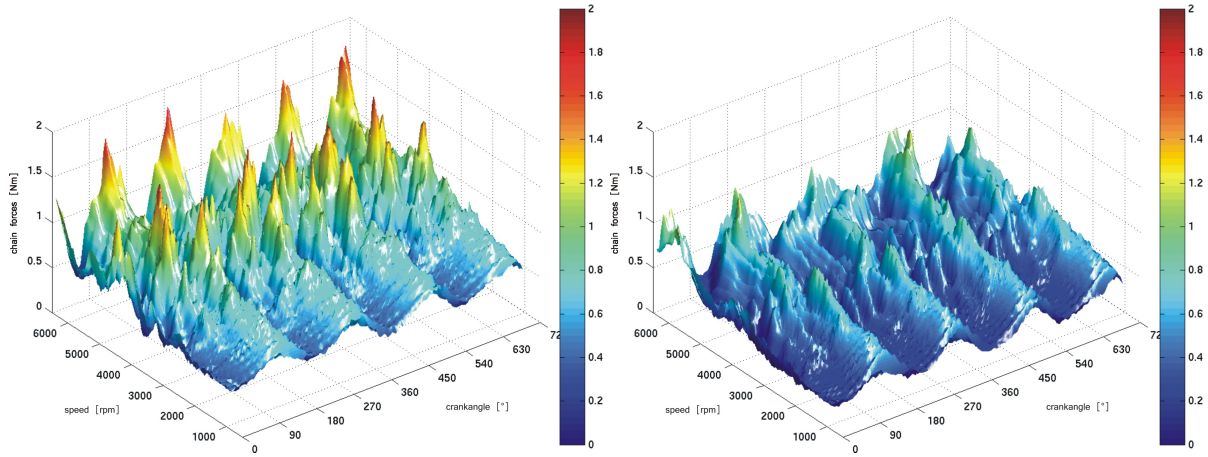


Figure 2: chain forces before and after optimization

6 Conclusion and Future Development

Surrogate-based algorithms are an appreciated tool for single-criteria optimization with time-consuming objectives. Extended to solve multiobjective optimization problems they provide a possibility to obtain an approximation of the Pareto set. A minimal number of evaluations of the objective is possible at the price of an increased complexity of the surrogate model.

We would like to raise two questions for further development of surrogate-based multiobjective optimization. Firstly, research is necessary to identify reliable criteria for exploiting the surrogate-model at more than one location. By differentiating between depth-first and breadth-first search we can compute at least some evaluations of the objective function in parallel. With regard to the increased number of processors in every computer it will be advantageous to make use of parallelization in optimization algorithms. Furthermore, the run time of construction and training of surrogate-models should be decreased to expand their potential of application.

References

- Kalyanmoy Deb, Samir Agrawal, Amrit Pratap, and T. Meyarivan. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: Nsga-ii, 2000.
- Donald R. Jones, Matthias Schonlau, and William J. Welch. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13:455 – 492, 1998.
- Joshua Knowles. Parego: A hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems, 2006.