Optimization by Gaussian Processes assisted Evolution Strategies

Holger Ulmer, Felix Streichert, and Andreas Zell

Center for Bioinformatics Tübingen (ZBIT), University of Tübingen,
Sand 1, 72074 Tübingen, Germany
ulmerh,streiche,zell@informatik.uni-tuebingen.de,
http://www-ra.informatik.uni-tuebingen.de

Abstract. Evolutionary Algorithms (EA) are excellent optimization tools for complex high-dimensional multimodal problems. However, they require a very large number of problem function evaluations. In many engineering optimization problems, like high throughput material science or design optimization, a single fitness evaluation is very expensive or time consuming. Therefore, standard evolutionary computation methods are not practical for such applications. Applying models as a surrogate of the real fitness function is a quite popular approach to handle this restriction. We propose a Model Assisted Evolution Strategy (MAES), which uses a Gaussian Process (GP) approximation model. The purpose of the Gaussian Process model is to preselect the most promising solutions, which are then actually evaluated by the real problem function. To refine the preselection process the likelihood of each individual to improve the overall best found solution is determined. Numerical results from extensive simulations on high dimensional test functions and one material optimization problem are presented. MAES has a much better convergence rate and achieves better results than standard evolutionary optimization approaches with less fitness evaluations.

1 Introduction

Evolution Strategies (ES) are one class of Evolutionary Algorithms (EAs) which are often used as optimization tools for complex high dimensional multimodal problems [8] [9]. In contrast to other EAs like Genetic Algorithms or Genetic Programming ES work directly on real valued objective variables, which represent a possible solution. Therefore ES are very suitably for many engineering optimization problems.
However, like other population based EAs ES require a very high number of fitness function evaluations to determine an acceptable solution. In most real world engineering optimization applications the process of fitness evaluation is very expensive and time consuming. Therefore standard ES methods are not practical for such applications.
A promising approach to handle this problem is the application of modeling techniques, where a model evaluation is orders of magnitude cheaper than

a real fitness function evaluation. A model is trained on already evaluated fitness cases and is used to guide the search for promising solutions. This approach decreases the number of expensive fitness evaluations and has a better convergence rate. The application of modeling techniques in evolutionary computation receives increasing attention [7] [2] [3] [10]. A survey on this research field can be found in [6].
The remainder of this paper is organized as follows: Section 2 introduces the synthesis of the Gaussian Process (GP) fitness approximation model with a standard ES. The GP model is utilized to assist the ES by selecting the most promising solutions of an offspring population to be evaluated by the real fitness function. Numerical results from extensive simulations on high dimensional artificial test functions and one material optimization task are presented and discussed in section 3. The paper closes with a brief conclusion and outlook on future work.

2 GP Model Assisted Evolution Strategy

For the approximation of the fitness function we chose Gaussian Processes (GP), which are general and proper real valued function approximators, especially for noisy training data. A detailed description is given in [4]. Compared to other models like artificial neural networks GP are probabilistic models, which have the advantage of providing a confidence value given by the standard deviation σ for the predicted fitness value t without additional computational cost. Moreover GP are stable against overfitting and have only a limited number of model parameters, which have to be chosen by the user.
[9]. We start our consideration with a standard (μ, λ) ES, which will be later coupled with the GP model. An ES works on a population of potential solutions $\boldsymbol{x}$ (individuals) by manipulating these individuals with evolutionary operators [8]. By applying the evolutionary operators reproduction, recombination and mutation (see pseudocode in Figure 1) λ offspring individuals are generated from μ parents. After evaluating the fitness of the λ offspring individuals, μ individuals with the best fitness are selected by a (μ, λ) strategy to build the parent population for the next generation. The algorithm terminates when a maximum number of fitness function evaluations have been performed.
To incorporate the approximation model into the ES we use a pre-selection concept similar to the one described by Emmerich et al. [3]. Compared to the standard ES $\lambda_{Pre} > \lambda$ new offspring individuals are created from μ parents (see pseudocode in Figure 2). These λ_{Pre} individuals have to be pre-selected to generate the offspring of λ individuals, which will be evaluated with the real fitness function. The model is trained at the beginning with a randomly created initial population and is updated after each generation step with λ new fitness cases. The key point of our approach is the pre-selection procedure. Using the mean of the model prediction to identify the most promising

```
Procedure ES
Begin
 eval=0;
 Pop=CreateInitialPop();
 Pop.EvaluateRealFitness();

 while (eval<maxeval);
  Offspring=Pop.Reproduce(λ);
  Offspring.Mutate();


  Offspring.EvaluateRealFitness();

  Pop=Offspring.SelectBest(μ);
  eval=eval+λ;
 end while
End
```

Fig. 1. Standard (μ, λ) Evolution Strategy (ES).

```
Procedure MAES
Begin
 eval=0;
 Pop=CreateInitialPop();
 Pop.EvaluateRealFitness();
 Model.update(Pop);
 while (eval<maxeval);
  PrePop=Pop.Reproduce(λ_Pre);
  PrePop.Mutate();
  PrePop.EvaluateWithModel();
  Offspring=PrePop.SelectBest(λ);
  Offspring.EvaluateRealFitness();
  Model.update(Offspring);
  Pop=Offspring.SelectBest(μ);
  eval=eval+λ;
 end while
End
```

Fig. 2. Model Assisted Evolution Strategy (MAES).

individuals leads to premature and suboptimal convergence rate on multimodal problems with many misleading local minima, because individuals with a better model prediction are preferred to others and therefore have a lower probability to escape from these minima.

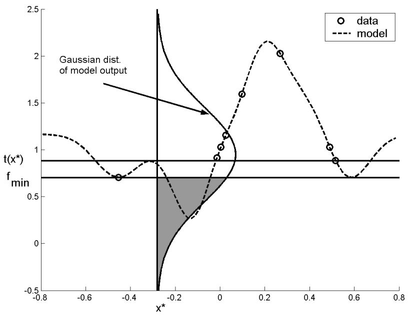

Fig. 3. The gray filled area represents the probability that a model output value t is sampled at point $x^\star$, which is smaller than f_{min} (POI).

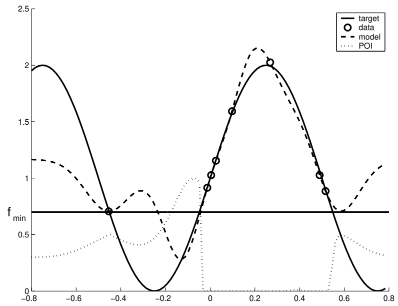

Fig. 4. Areas with a higher POI criterion have a higher probability to sample a data point with a target value smaller than f_{min}.

To address this problem we use a new pre-selection criterion, which utilizes the model confidence given by the GP model as the standard deviation $\sigma(\boldsymbol{x})$.

The idea is not new in the field of global optimization [1], but new in the context of evolutionary optimization. The concept is illustrated in Figure 3. At any given point $\boldsymbol{x}$, we model the uncertainty about the model value prediction by considering this value to be like the realization of a random variable $Y(\boldsymbol{x})$ with mean $\widehat{t}(\boldsymbol{x})$ and standard deviation $\sigma(\boldsymbol{x})$.
Let $f_{min} = min(t_1, ..., t_N)$ be the current best fitness value sampled until now, then the target value for the improvement will be some number $T \leq f_{min}$. The Probability Of Improvement (POI) is simply the probability that $Y(\boldsymbol{x}) \leq T$. Assuming the random variable is normal distributed, this probability is given by

$$POI(\boldsymbol{x}) = \Phi\left(\frac{T - \widehat{t}(\boldsymbol{x})}{\sigma(\boldsymbol{x})}\right) \tag{1}$$

where $\Phi(\cdot)$ is the normal cumulative distribution function. Figure 4 shows the characteristics of the POI selection criterion. Areas with a high POI have a high probability to sample a data point with a target value smaller than f_{min} and are therefore more promising. Areas with model prediction $\widehat{t}(\boldsymbol{x}) >> f_{min}$ have a low probability of improvement $POI \approx 0$. As the function is sampled more and more around the current best point, the standard deviation in this area decreases. The term $\frac{T-\widehat{t}(\boldsymbol{x})}{\sigma(\boldsymbol{x})}$ becomes extremely negative and POI will be so small that the algorithm is driven to search elsewhere in unexplored areas where the standard deviation is higher. Therefore POI prefers unexplored areas of object space and has a multimodal characteristic. Note, that the maximal POI value may have another location in object space than the minimal model output value. The individuals $\boldsymbol{x}_i$, $i = 1, .., \lambda_{Pre}$ with the highest $POI(\boldsymbol{x})$ are pre-selected to build the new offspring.

The size of the pre-selected population λ_{Pre} controls the impact of the model on the evolutionary optimization process. For $\lambda_{Pre} = \lambda$, the algorithm performs like a standard (μ, λ) ES. Increasing λ_{Pre} results in a larger selection pressure in the pre-selection and in a stronger impact of the model on the convergence behavior of the optimization process.

3 Experimental results and discussion

To analyze the algorithms extensive numerical simulations were performed for 5 artificial test functions and one material optimization problem. For each case the standard (μ, λ)-ES algorithm is compared with the new (μ, λ)-MAES algorithm for population size $(\mu = 5, \lambda = 35)$.
The size of the pre-selected population λ_{Pre} was set to 3λ. We used Covariance Matrix Adaption (CMA) developed by Hansen et al.[5], which is a powerful method for adaption of the mutation step size. For all simulations no recombination was used and the initial population size was set to 10.

The training data for the GP model consisted of the 2λ most recently performed fitness evaluations. For this reason the model is a local model of the individual's neighborhood in object space. Using more training data improves the performance only slightly but results in with much higher computational costs for model training.
The values are always evaluated as the mean of 100 repeated runs with different seed values for random number generation.
The Sphere function $f_{Sphere}(\boldsymbol{x}) = \sum_{i=1}^{20} x_i^2$ is a nonlinear and unimodal test function which is a good test for the self-adaptation mechanism of ES.
The MAES (see Figure 5) shows a better convergence rate and outperforms the standard ES clearly. It reaches 10 times better fitness values after 5000 evaluations. Comparable results are obtained with several other unimodal functions. In Figure 6 results are presented for the Schwefel 1.2 test function: $f_{Schwefel}(\boldsymbol{x}) = \sum_{i=1}^{20} \left(\sum_{j=1}^{i} x_j \right)^2$.
The Rosenbrock function (2) is nonlinear, continuous and not symmetric.

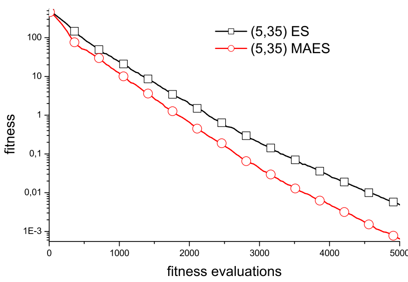

Fig. 5. 20-dim. Sphere function: fitness of best individual.

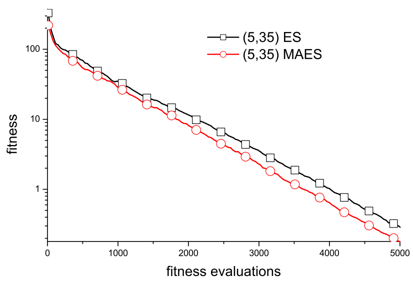

Fig. 6. 20-dim. Schwefels's function 1.2: fitness of best individual.

$$f_{Rosen}(\boldsymbol{x}) = \sum_{i=1}^{20} (100 \cdot (x_{i+1} - x_i)^2 + (x_i - 1)^2) \qquad (2)$$

It is a very popular test function and has a very hard to find global optimum. Figure 7 shows that MAES reaches better fitness values than the standard ES. These results justifies the motivation of the model assisted approach to support the algorithm by identifying the most promising individuals.
Multimodal functions evoke hills and valleys, which are misleading local optima. A simple optimization algorithm like hill-climbing would get stuck in a local minimum. For such problems evolutionary algorithms are much more appropriate.

Ackley's test function (3) is symmetric and very bumpy.

$$f_{Ack}(\boldsymbol{x}) = 20 + e - 20\exp\left(-0.2 \cdot \sqrt{\frac{1}{20} \cdot \sum_{i=1}^{20} x_i^2}\right) - \exp\left(\frac{1}{20}\sum_{i=1}^{20}\cos(2\pi x_i)\right) \tag{3}$$

Its number of local minima increases exponentially with the problem dimension and has a global optimum with very strong local features. Here MAES converges faster than the standard ES (see Figure 8). Comparable results are obtained for Rastrigin's test function (3) (see Figure 9).

$$f_{Rast}(\boldsymbol{x}) = 10 \cdot 20 + \sum_{i=1}^{20}(x_i^2 - \cos(2\pi x_i)) \tag{4}$$

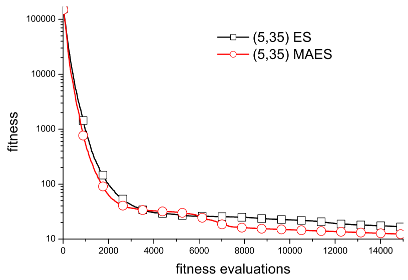

Fig. 7. 20-dim. Rosenbrock function: fitness of best individual.

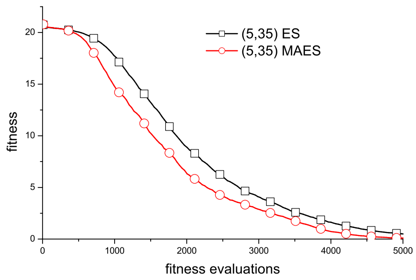

Fig. 8. 20-dim. Ackley's function: fitness of best individual.

The last presented result shows the application of the new MAES algorithm on a real world engineering optimization. Here the task is to optimize certain chemical catalytic properties of solid state samples created by a high throughput process. The objective function of the problem is modeled by a artificial neural network with data from already performed experiments and depends on 6 input variables, which describe the synthesis of the samples. The MAES algorithm outperforms the standard ES (see Figure 10). From the beginning MAES has a higher convergence rate and yields better solutions. It clearly yields better solutions with less fitness function evaluations. For example MAES reaches 2 times better solutions after 400 evaluations than the standard ES. Therefore MAES halves the costs of optimization for this application.

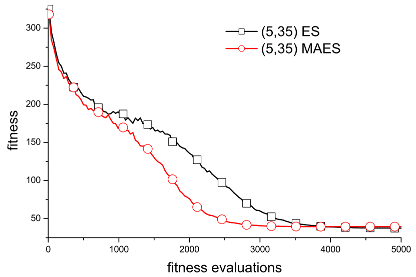

Fig. 9. 20-dim. Rastrigin's function: fitness of best individual.

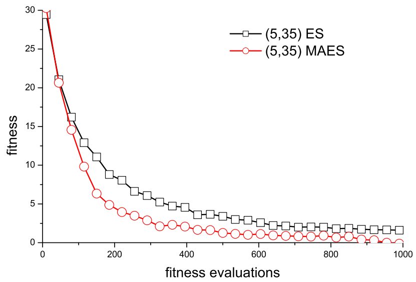

Fig. 10. 6-dim. Material optimization: fitness of best individual.

4 Conclusions

We applied a Gaussian Process as an approximation model to assist a standard ES by using the GP to pre-select the most promising individuals to be evaluated by the real fitness function.

The pre-selection procedure is given by the Probability Of Improvement (POI) pre-selection criterion. POI addresses the tradeoff between exploitation and exploration by utilizing the probabilistic interpretation of the GP model. This is done by evaluating the likelihood of each individual to improve the overall best solution.

Extensive simulations on artificial test functions showed that this approach enhances the performance of a standard ES on unimodal and multimodal problems. MAES has a higher convergence speed and is much more stable against premature convergence for multimodal problems. This is reasonable, because the MAES with POI pre-selection criterion has a higher tendency to sample in unexplored areas.

For a material optimization problem MAES yields the same solution quality with half the function evaluations and achieves overall better solutions. Therefore the application of MAES halves the costs compared to the standard ES method. These encouraging results justify the application of MAES in the field of engineering optimization applications where problem evaluations are very costly.

For further work it is planned to develop a mechanism which controls the impact of the approximation model on the optimization process by controlling λ_{Pre}. This can be carried out by using the confidence of the approximation model.

Acknowledgments This research has been funded by the German federal ministry of research and education (BMBF) in the project "Entwicklung eines Systems zur automatisierten Herstellung und Charakterisierung von kristallinen Festkörpern in hohem Durchsatz" under contract No. 03C0309E.

References

1. M. Schonlau D. Jones and W. Welch. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13:455–492, 1998.
2. M. A. El-Beltagy and A. J. Keane. Evolutionary optimization for computationally expensive problems using gaussian processes. In CSREA Press Hamid Arabnia, editor, *Proc. Int. Conf. on Artificial Intelligence IC-AI'2001*, pages 708–714, 2001.
3. M. Emmerich, A. Giotis, M.Multlu Özdemir, T. Bäck, and K. Giannakoglou. Metamodel-assisted evolution strategies. In *Parallel Problem Solving from Nature VII*, pages 362–370, 2002.
4. M. Gibbs and D. MacKay. Efficient implementation of gaussian processes. Technical report, Cavendish Laboratory, Cambridge, UK, 1997.
5. N. Hansen and A. Ostermeier. Convergence properties of evolution strategies with the derandomized covariance matrix adaptation: The $(\mu/\mu_i, \lambda)$-cma-es. In *5th European Congress on Intelligent Techniques and Soft Computing*, pages 650–654, 1997.
6. Y. Jin. A comprehensive survey of fitness approximation in evolutionary computation. *Soft Computing Journal*, 2003. In press.
7. Y. Jin, M. Olhofer, and B. Sendhoff. A framework for evolutionary optimization with approximate fitness functions. *IEEE Transactions on Evolutionary Computation. March 2002 (in press). (ISSN: 1089-778X)*, 2002.
8. I. Rechenberg. *Evolutionsstrategie '94*. frommann-holzboog, Stuttgart, 1994.
9. H.-P. Schwefel. *Numerische Optimierung von Computer-Modellen mittels der Evolutionsstrategie*. Birkhäuser, Basel, 1977.
10. H. Ulmer, F. Streichert, and A. Zell. Model-assisted steady-state evolution strategies. In *GECCO 2003 Proceedings of Genetic and Evolutionary Computation Conference*, pages 610–621, 2003.