A Mixed Bayesian Optimization Algorithm with variance adaptation

Jiri Ocenasek[1], Stefan Kern[2], Nikolaus Hansen[2], Petros Koumoutsakos[1,2]

[1] Computational Laboratory (CoLab), Swiss Federal Institute of Technology ETH
[2] Institute of Computational Science, Swiss Federal Institute of Technology ETH
Hirschengraben 84, 8092 Zürich, Switzerland,
{jirio,skern,hansenn,petros}@inf.ethz.ch

Abstract. This paper presents a hybrid evolutionary optimization strategy combining the Mixed Bayesian Optimization Algorithm (MBOA) with variance adaptation as implemented in Evolution Strategies. This new approach is intended to circumvent some of the deficiences of MBOA with unimodal functions and to enhance its adaptivity. The Adaptive MBOA algorithm – AMBOA – is compared with the Covariance Matrix Adaptation Evolution Strategy (CMA-ES). The comparison shows that, in continuous domains, AMBOA is more efficient than the original MBOA algorithm and its performance on separable unimodal functions is comparable to that of CMA-ES.

1 Introduction

A class of Evolutionary Algorithms (EAs) implement probability distributions to identify the underlying relationship of the objective function with its parameters in order to accelerate the convergence rate of the algorithms. Estimation of Distribution Algorithms (EDAs) [1–3] sample a probability distribution learned from the fittest solutions. A class of EDAs use a Bayesian network with a local structure in the form of a decision graph to model the relationship between discrete parameters on a global level. In addition to that, the Mixed Bayesian Optimization Algorithm (MBOA) [4] is able to deal with discrete and continuous parameters simultaneously by using a Gaussian kernel model to capture the local distribution of the continuous parameters.

MBOA has been shown to perform successfully for several combinatorial problems [5]. However, on certain continuous benchmarks – including unimodal functions – MBOA is outperformed [6] by Evolution Strategies, like Covariance Matrix Adaptation ES (CMA-ES) [7, 8]. The reason for this is attributed to the relative deficiency of MBOA in adapting the variance of the search distribution. In order to overcome this deficiency we propose a new variance adaptation mechanism which significantly increases the efficiency of MBOA in continuous domains.

Section 2 introduces the principles of the MBOA algorithm. In Section 3 we analyze the main difference between the estimation of variance in MBOA and in CMA-ES and propose an improved algorithm, AMBOA, with robust adaptation

of the estimated variance. In Section 4 we present experimental results that demonstrate the successful design of AMBOA.

2 Mixed Bayesian Optimization Algorithm (MBOA)

2.1 Main principles of MBOA

MBOA belongs to the class of Estimation of Distribution Algorithms (EDAs) that explore the search space by sampling a probability distribution that is developed during the optimization. MBOA works with a population of N candidate solutions. Each generation, typically the $N/2$ fittest individuals are used for the model building and $N/2$ new solutions are generated from the model. These offspring individuals are evaluated and incorporated into the original population, replacing some of the old ones. This process is repeated until the termination criteria are met.

A Bayesian network (BN) is one of the general models to express discrete probability distributions. The underlying probability distribution $p(\boldsymbol{X})$ is approximated as the product of conditional probability distributions of each parameter X_i given $\boldsymbol{\Pi}_i$ – the variables that influence X_i

$$p(X_0, ..., X_{n-1}) = \prod_{i=0}^{n-1} p(X_i|\boldsymbol{\Pi}_i). \tag{1}$$

We use upper case symbols X_i to denote the i-th design parameter (or the i-th gene in EA terminology or the i-th random variable in mathematical terminology) whereas lower-case symbols x_i denote a realization of this parameter. Boldface symbols distinguish vectors from scalars. $\boldsymbol{x}_j^{(g)}$ denotes the j-th individual in generation number g.

The construction of an optimal BN from the population of candidate solutions is itself an NP hard problem [9], and EDAs usually use either an incremental or a greedy version of the learning algorithm to accelerate the BN construction. MBOA uses the latter approach.

MBOA can be formulated for continuous and discrete domains. In binary domain it performs similarly to the hierarchical Bayesian Optimization Algorithm [10], but it employs a different model building algorithm which ensures efficient parallelization. In continuous domains, MBOA searches for a decomposition of the search space into partitions where the parameters seem to be mutually independent. This decomposition is captured globally by the Bayesian network model and Gaussian kernel distributions are used locally to approximate the values in each resulting partition.

2.2 Construction of the continuous probabilistic model in MBOA

MBOA attempts to capture the local conditional probability density functions of the continuous parameters $f(X_i|\boldsymbol{\Pi}_i)$. Each $f(X_i|\boldsymbol{\Pi}_i)$ is captured in the form of a

decision tree [11], which is more efficient than the traditional way of keeping $\boldsymbol{\Pi}_i$ explicitly in the form of a dependency graph and using tabular representations for local conditional distributions.

We will describe how the decision tree for a concrete parameter X_i is constructed from the population D. In particular, for each parameter X_i the set of influencing variables $\boldsymbol{\Pi}_i$ has to be determined. Since it is computationally expensive to test independence directly in the continuous domain MBOA recursively transforms the problem into binary domain.

First, X_i and all continuous parameters that are available as the potential candidates to form $\boldsymbol{\Pi}_i$ are temporarily converted into new binary variables by defining continuous split boundaries on them. The method for finding the boundaries is presented in [4]. As soon as all variables are discrete, the Bayesian-Dirichlet metrics with likelihood equivalence (BDe) [12] is used to determine the variable that influences X_i the most. The chosen variable is then used for splitting the population D and the construction is recursively repeated for both branches of the new split. The recursion stops when for all variables the BDe score (decreased by the complexity penalty term) returns a negative value.

This results in a decomposition of the $f(X_i|\boldsymbol{\Pi}_i)$ domain into axis-parallel partitions where X_i is assumed to be decorrelated from the variables in $\boldsymbol{\Pi}_i$ and can be approximated by univariate probability density functions. The Gaussian kernel distribution of a parameter X_i in a concrete leaf j given a concrete π_i (the realization of $\boldsymbol{\Pi}_i$) can be expressed as:

$$f(X_i|\boldsymbol{\pi}_i \in \{\boldsymbol{\pi}_i\}_j) = \frac{1}{|\{x_i\}_j|} \sum_{\forall m \in \{x_i\}_j} \mathcal{N}(m, \sigma_{ij}^2) \qquad i = 0, ..., n-1, \tag{2}$$

where $\{\boldsymbol{\pi}_i\}_j$ denotes the set of all possible realizations of $\boldsymbol{\Pi}_i$ traversing to the j-th leaf, $\{x_i\}_j \subset \mathbb{R}$ denotes the set of realizations of variable X_i among the individuals from population D that traverse to j-th leaf, and $|\{x_i\}_j|$ denotes the size of this set. All the kernels in the same leaf have the same height $1/|\{x_i\}_j|$ and the same width σ_{ij}. In our experiments we set σ_{ij} equal to

$$\sigma_{ij} = \frac{\max\{x_i\}_j - \min\{x_i\}_j}{r}, \tag{3}$$

where the default setting for the scaling factor r in MBOA is $r = |\{x_i\}_j| - 1$.

The newly generated offspring population is used to replace some part of the former population. For effective diversity preservation, MBOA uses the so-called Restricted Tournament Replacement (RTR). In RTR, each offspring competes with the closest individual selected from a random subset of the former population. This subset comprises 5% of the population in our experiments.

3 Adaptive MBOA - AMBOA

3.1 Motivation

We investigated the susceptibility of the original MBOA to premature convergence and compared it to the Evolution Strategy with Covariance Matrix Adap-

Name	Function	Stop. criterion
Plane	$f_{\text{plane}} = -x_0$	-10^{10}
Sphere	$f_{\text{sphere}} = \sum_{i=0}^{n-1} x_i^2$	10^{-10}
Ellipsoid	$f_{\text{elli}} = \sum_{i=0}^{n-1} 10^{4\frac{i}{n-1}} x_i^2$	10^{-10}
Rastrigin	$f_{\text{Rastrigin}} = 10n + \sum_{i=0}^{n-1}(x_i^2 - 10\cos(2\pi x_i))$	10^{-10}

Table 1. Test functions to be minimized and the corresponding stopping criterion. The initial solutions were generated uniformly using the initialization region $[0.5, 1.5]^n$ for f_{plane} and $[-3, 7]^n$ for the other functions, the global step size $\sigma^{(g)}$ of CMA-ES was initialized to 1.0 for f_{plane} and 5.0 for the other functions.

tation [7, 8]. We used the (μ, λ)-CMA-ES as described in [13], where the covariance matrix $\boldsymbol{C}$ is updated by μ ranked parents selected from the λ individuals.

The ability to enlarge the overall population variance can be tested using the plane function f_{plane} (see Tab. 1). Within a small enough neighborhood, a linear function is a good approximation for any smooth function. Therefore, f_{plane} is a good test case for a situation where the population variance is (far) too small. Fig. 1a shows the function value versus the number of function evaluations for both methods. It can be seen that CMA-ES reaches the termination criteria of f_{plane} fast, using a population of only 10 individuals. On the other hand, MBOA is slower by 3 orders of magnitude. The reason is that MBOA – unlike CMA-ES – has no effective mechanism to increase the variance. Up to large population sizes ($N < 3200$ for $n = 10$ on f_{plane}), MBOA is not able to divert the solutions to fulfill the stopping criterion. We observed that the variance shrinks faster than the mean of the distribution moves. With $N \geq 3200$, the Restricted Tournament Replacement (RTR) is able to reduce shrinking, but at the expense of slow convergence.

Subsequently, we tested MBOA on the sphere function (see Tab. 1) and increased the population size according to the sequence 10, 20, 50, 100, 200, 400, 800, 1600, 3200 until the optimum was found in all 20 runs. In Fig. 1b it is shown that population size N=100 is needed by MBOA to solve the 10-dimensional f_{sphere} benchmark. With lower population size some of MBOA runs were not able to reach the precision 10^{-10}.

Consequently, we identified that MBOA performance is harmed in case of low population size. This can be explained by the large deviation present in the estimated parameters. These deviations are further propagated by iterated sampling and re-estimation. In contrast, CMA-ES adjusts the variance robustly and needs only very small populations. Moreover, the model parameters in CMA-ES are updated incrementally, which makes them less susceptible to deviations.

3.2 Design of AMBOA

Based on the above observations, we aimed at improving MBOA. To prevent variance from premature shrinking, we experimented with the width of Gaussian

kernel σ_{ij} by setting the factor r (in Eq. (3)) as $r = \sqrt{|\{x_i\}_j|}$ or even $r = 1$. However, different benchmarks required different settings to perform efficiently. Therefore, we introduce an overall scaling factor, η, to control the kernel width of the marginal distributions adaptively:

$$f(X_i|\boldsymbol{\pi}_i \in \{\boldsymbol{\pi}_i\}_j) = \frac{1}{|\{x_i\}_j|} \sum_{\forall m \in \{x_i\}_j} \mathcal{N}(m, {\eta^{(g)}}^2 \sigma_{ij}^2). \tag{4}$$

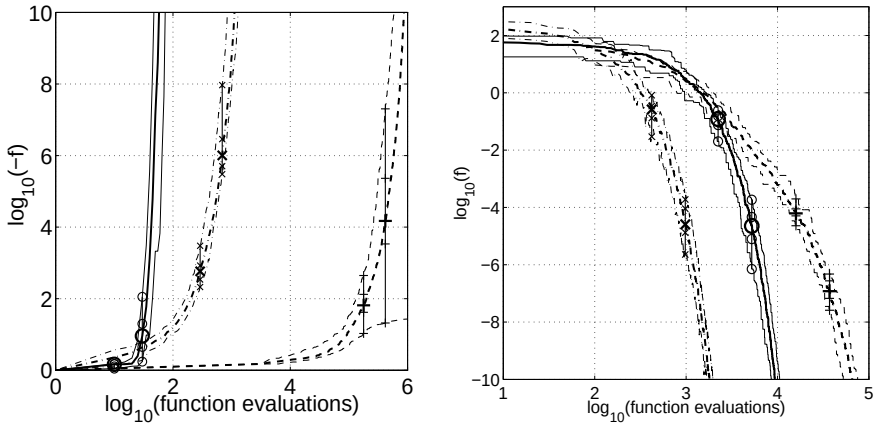
Compared to eq. (2) one can see that the factor η is used to scale the width of each kernel. Inspired by the well-known 1/5-success rule for ESs [14], the factor is adjusted according to the success rate of RTR. In our implementation the information about success or failure of each individual is immediately accumulated into the step size. In case of success the factor is multiplied by α, otherwise it is multiplied by $\alpha^{\frac{p}{p-1}}$. For $N/2$ offspring individuals (with N_{succ} successes and N_{fail} failures), the total change of factor in the g-th generation can be expressed as

$$\eta^{(g+1)} = \eta^{(g)} \alpha^{N_{succ}} \alpha^{N_{fail} \frac{p}{p-1}}, \tag{5}$$

where p denotes the desired success rate (for $N_{succ}/(N_{succ} + N_{fail}) = p$ it holds $\eta^{(g+1)} = \eta^{(g)}$). The choice of α determines how fast the desired success rate is achieved. For increasing α the adaptation is faster, but also more sensitive to oscillations. Our experiments with f_{sphere} and $f_{\text{Rastrigin}}$ indicate that the choice of α does not depend significantly on the problem size n, but it depends on the population size N. To limit the maximal change of $\eta^{(g+1)}$ per generation, we choose $\alpha = e^{4/N}$. If all the offspring individuals are accepted - which is very unlikely – then it holds $\eta^{(g+1)} = e^2 \eta^{(g)}$. We also performed a number of experiments to determine the optimal choice of p. For unimodal functions the choice of p is not critical (2a) and the Rechenberg's rule $p = 1/5$ could have been used. For several multimodal functions the optimal p is decreasing with problem size. This is demonstrated in Fig. 2b for $f_{\text{Rastrigin}}$. As the trade-off between speed of solving unimodal test functions and robustness of solving multimodal test functions, we choose $p = 0.05 + \frac{0.3}{\sqrt{n}}$. Detailed analysis of the proper choice of a success rate for deceptive functions and the role of RTR during the adaptation will be a subject of future research.

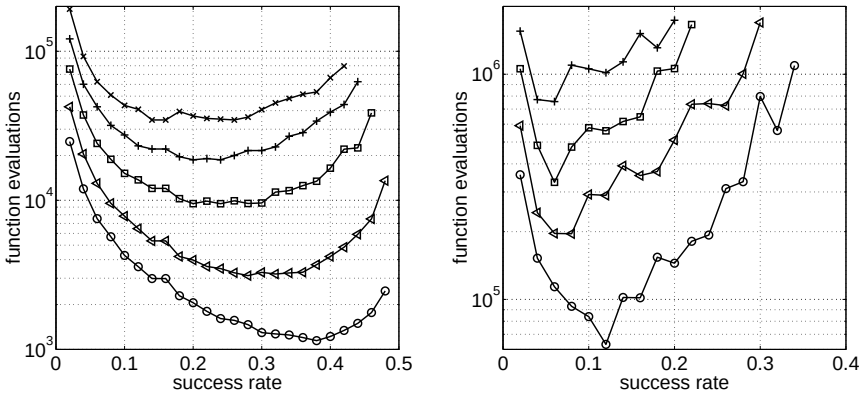
4 Experimental results

We compare the performance of the newly proposed AMBOA to the original MBOA and to CMA-ES. The benchmark functions are summarized in Tab. 1. All functions are separable, and only $f_{\text{Rastrigin}}$ is multimodal. In Fig. 1, 3, and 4 the bold lines are the median values of 20 runs, whereas thin lines show the minimum and maximum values. The five symbols per each measurement represent maximum, 75-percentile, median, 25-percentile, and minimum function values. The plots show results for the minimal population size for which all 20 runs converged. We start each experiment with population size $N = \lambda = 10$. If any of



(a) (b)

Fig. 1. Function value versus the number of function evaluations for MBOA (dashed line,'+'), CMA-ES (dot-and-dashed line,'×') and AMBOA (solid line,'∘') on 10-dimensional f_{plane} (a) and f_{sphere} (b). Population sizes (a): $\lambda = 10$ for CMA-ES, $N = 3200$ for MBOA and $N = 10$ for AMBOA. Population sizes (b): $\lambda = 10$ for CMA-ES, $N = 100$ for MBOA, $N = 10$ for AMBOA. The five symbols per each measurement represent maximum, 75-percentile, median, 25-percentile, and minimum function values of 20 runs.



(a) (b)

Fig. 2. (a) The influence of chosen success rate p on the number of evaluations MBOA needs to solve 5-dimensional ('∘'), 10-dimensional ('◁'), 20-dimensional ('□'), 30-dimensional ('+') , and 40-dimensional ('×') f_{sphere}. (b) The influence of chosen success rate p on the number of evaluations MBOA needs to solve 15-dimensional ('∘'), 25-dimensional ('◁'), 35-dimensional ('□'), and 45-dimensional ('+') $f_{\text{Rastrigin}}$. The success rates from $p = 0.02$ to $p = 0.48$ in 0.02 steps were examined. Median values out of 20 runs are shown for success rates where MBOA converged within less than $2e10^6$ fitness evaluations in at least 50% cases. Population size (a) $N = 10$, (b) $N = 100$.

20 runs do not reach the convergence criterion, the population size is increased according to the sequence 10,20,50,100,200,400,800,1600,3200 until the method converges in all 20 runs. The maximum population size tested was 3200.

We test AMBOA on f_{plane}, and f_{sphere}. In Section 3.1 these functions appeared to be difficult for MBOA. AMBOA effectively increases the variance on the 10-dimensional f_{plane} as shown in Fig. 1a. In addition, it requires a significantly smaller population size of $N = 10$ compared to $N = 3200$ for MBOA and it even requires less fitness evaluations than CMA-ES. AMBOA needs only a population size of $N = 10$ to reliably optimize the 10-dimensional f_{sphere} function, whereas MBOA needs $N = 100$ (Fig. 1b). The variance adaptation mechanism decreases the minimal required population size. This results in lower number of fitness evaluations, proportionally to the decrease of N. The same type of AMBOA behaviour is evident from Fig. 3, where the results of optimizing the 10-dimensional f_{elli} are depicted. The comparison of results from Fig. 1b and 3 indicates that AMBOA performs similarly on f_{sphere} and f_{elli}. The adaptation of the scaling factor η plays the same role in both cases, whereas the relative scaling of the individual coordinates is estimated empirically. In contrast, for CMA-ES it is much easier to adapt on the sphere function, because it starts with the spherically shaped distribution (so it is sufficient to adapt the mean and scaling factor only), whereas for the f_{elli} it has to estimate the complete covariance matrix.

We compare AMBOA, MBOA and CMA-ES on the 10-dimensional Rastrigin function (Fig. 4). The Rastrigin function is multimodal but its underlying model is a hyper-paraboloid. With a large population size $\lambda = 800$ CMA-ES is able to discover this underlying model in all 20 runs. AMBOA needs a population size of $N = 100$ whereas MBOA needs $N = 200$. With smaller population sizes the algorithms get stuck in a local optimum. Since AMBOA does not approximate the fitness landscape by a unimodal density, there is a different way how AMBOA explores the search space. We assume that AMBOA and MBOA utilize RTR to keep samples from the neighborhood of several local minima. Since the problem is separable, the local minima in all dimensions are sampled independently to form new solutions. Provided that in each dimension there is at least one solution that contains the proper value, the global optimum is reached after a small number of trials. The slope of the convergence curve of AMBOA is steeper than that of MBOA. This indicates that the variance adaptation plays a role in the local improvement of new solutions.

We investigate how AMBOA and CMA-ES behave for an increasing number of dimensions on f_{elli} (Fig. 5). We measure the number of fitness evaluations to achieve the given fitness in 2, 5, 10 and 20 dimensions, with $\lambda = 10$ for CMA-ES and $N = 10$ for AMBOA. The medians of 20 runs are shown. We observe that CMA requires less fitness evaluations to evolve high precision solutions, but the differences between AMBOA and CMA-ES decreases with increasing number of dimensions.

The proposed mechanism for variance adaptation allows MBOA to solve separable unimodal benchmarks with relatively small population sizes. With low

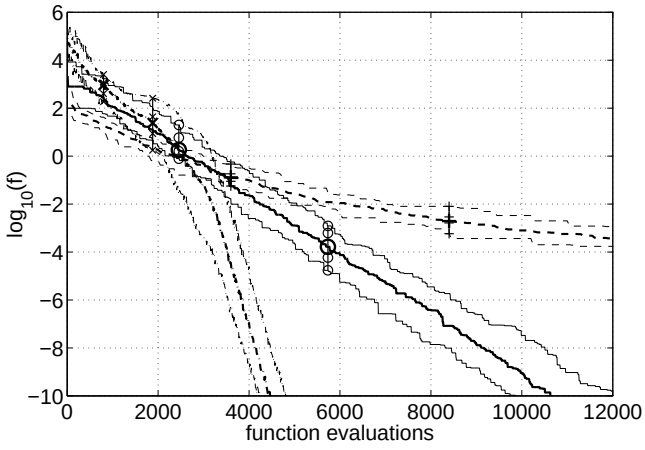


Fig. 3. AMBOA (solid line,'$\circ$'), CMA-ES (dot-and-dashed line,'$\times$'), and MBOA (dashed line,'+') on 10-dimensional f_{elli}. Population sizes: $\lambda = 10$ for CMA-ES, $N = 10$ for AMBOA and $N = 100$ for MBOA. The median of the number of required fitness evaluations to reach 10^{-10} precision was 4450 for CMA-ES, 5885 for AMBOA and 65650 for MBOA.

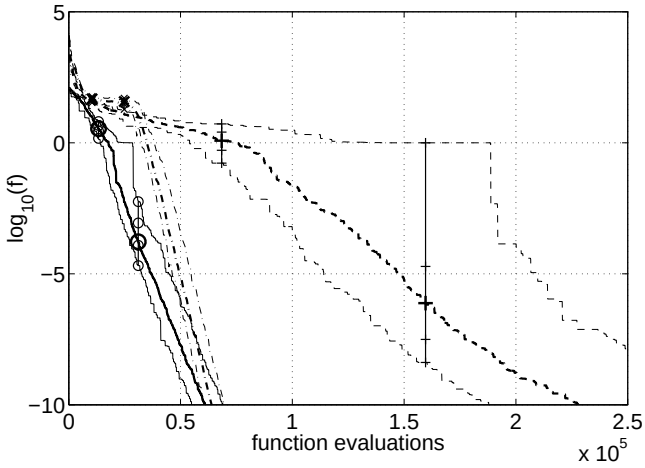


Fig. 4. AMBOA (solid line,'$\circ$'), CMA-ES (dot-and-dashed line,'$\times$') and MBOA (dashed line, '+') on 10-dimensional $f_{\mathrm{Rastrigin}}$. Population sizes: $\lambda = 800$ for CMA-ES, $N = 100$ for AMBOA and $N = 200$ for MBOA. The median of the number of required fitness evaluations to reach 10^{-10} precision was 38550 for AMBOA, 64000 for CMA-ES and 227900 for MBOA.

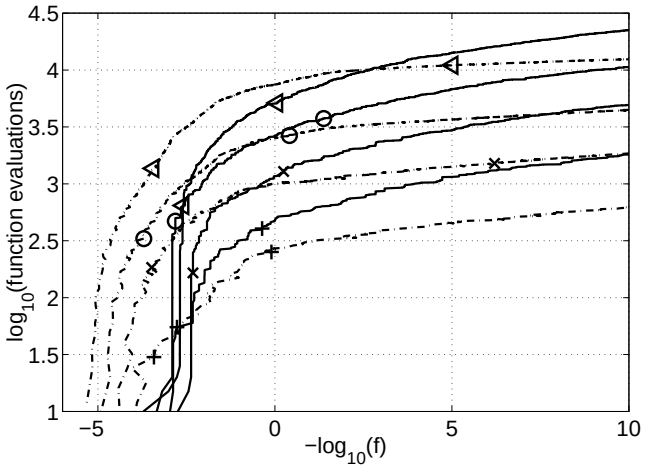


Fig. 5. Comparison of AMBOA (solid line) and CMA-ES (dashed line) behavior on 2-dimensional ('+'), 5-dimensional ('×'), 10-dimensional ('∘') and 20-dimensional ('◁') function f_{elli}. Population sizes: $\lambda = 10$ for CMA-ES and $N = 10$ for AMBOA.

population sizes the RTR behaves like the usual tournament replacement, so its niching effect is eliminated. Additionally, in case of small populations, MBOA penalizes most of the discovered dependencies and does not incorporate them into the model, thus imposing the separability of the optimized problem.

In case of nonseparable multimodal problems, our first experiments indicate that CMA performs better if the problem has a unimodal global underlying attractor, whereas AMBOA performs better for problems of combinatorial or deceptive nature.

5 Conclusion

Variance-adaptation is introduced to the Mixed Bayesian Optimization Algorithm as a necessary ingredient for the reliable and efficient solving of unimodal optimization problems. The newly proposed AMBOA algorithm uses a variance-adaptation mechanism based on the success rate of the replacement operator. The proposed mechanism can be also seen as an adaptation of Rechenberg's success rule for kernel-based distributions and can be used in general, not only within the scope of AMBOA. This approach does not rely on the assumption of unimodality and can be used together with the elitistic selection and replacement. On the examples of the separable test functions – plane, sphere, ellipsoid, and Rastrigin – we showed that the improved AMBOA performs comparably to Covariance Matrix Adaptation Evolution Strategy and requires a much lower population size and a much lower number of fitness evaluations than the original MBOA.

Most of the existing Estimation of Distribution Algorithms – for example the Iterated Density Estimation Evolutionary Algorithm [15] – do not have the means to effectively adjust the variance. The usefulness of the variance adaptation for the EDA framework and for non separable functions is a subject of future research.

References

1. Mühlenbein, H., Paass, G.: 1996, 'From Recombination of Genes to the Estimation of Distributions: I. Binary Parameters'. *Lecture Notes in Computer Science* **1141**, pp. 178–187, 1996.
2. Pelikan, M., Goldberg, D. E., Lobo, F.: 'A Survey of Optimization by Building and Using Probabilistic Models'. *IlliGAL Report No. 99018*, Illinois Genetic Algorithms Laboratory, University of Illinois at Urbana-Champaign, Urbana, IL, 1999.
3. Larrañaga, P.: 'A Review on Estimation of Distribution Algorithms'. In: P. Larrañaga and J. A. Lozano (eds.): *Estimation of Distribution Algorithms.* pp. 80–90, Kluwer Academic Publishers, 2002.
4. Ocenasek, J., Schwarz, J.: 'Estimation of Distribution Algorithm for mixed continuous- discrete optimization problems'. In: *2nd Euro-International Symposium on Computational Intelligance.* pp. 227–232, IOS Press, Kosice, Slowakia, 2002.
5. Schwarz, J., Ocenasek, J.: 'Bayes-Dirichlet BDD as a probabilistic model for logic function and evolutionary circuit decomposer'. In: *Proceedings of the 8th International Mendel Conference on Soft Computing, Mendel 2002*, Brno University of Technology, Brno, Czech Rep., pp. 117–124, 2002.
6. Kern, S., Hansen, N., Müller, S., Büche, D., Ocenasek, J., Koumoutsakos, P.: 'Learning Probability Distributions in Continuous Evolutionary Algorithms - Review and Comparison.' Natural Computing, **3** (1), pp. 77–112, 2004.
7. Hansen, N., Ostermeier, A.: 2001, 'Completely Derandomized Self-Adaptation in Evolution Strategies'. *Evolutionary Computation* **9**(2), pp. 159–195, 2001.
8. Hansen, N., Müller, S. D., Koumoutsakos, P.: 'Reducing the Time Complexity of the Derandomized Evolution Strategy with Covariance Matrix Adaptation (CMA-ES)'. *Evolutionary Computation* **11**(1), pp. 1–18, 2003.
9. Chickering, D.M., Geiger, D., Heckerman, D.E.: 'Learning Bayesian networks is NP-hard', *Technical Report MSR-TR-94-17*, Microsoft Research, Redmond, WA, 1995.
10. Pelikan, M., Goldberg, D. E., Sastry, K.: 'Bayesian Optimization Algorithm, Decision Graphs, and Occam's Razor', *IlliGAL Report No. 2000020*, University of Illinois at Urbana-Champaign, Illinois Genetic Algorithms Laboratory, Urbana, IL, 2000.
11. Friedman, N., Goldszmidt, M.: 'Learning Bayesian Networks with Local Structure', In: M. I. Jordan ed. *Learning and Inference in Graphical Models*, 1998.
12. Heckerman, D., Geiger, D., Chickering, M.: 'Learning Bayesian networks: The combination of knowledge and statistical data'. *Technical Report MSR-TR-94-09*, Microsoft Research, Redmond, WA, 1994.
13. Hansen, N., Kern, S.: 'Evaluating the CMA Evolution Strategy on Multimodal Test Functions'. *Parallel Problem Solving from Nature PPSN VIII*, Springer, Birmingham, 2004, accepted.
14. Rechenberg, I.: 'Evolutionsstrategie: Optimierung technischer System nach Prinzipien der biologischen Evolution,' Fromann-Holzboog, Stuttgart, 1973.
15. Bosman, P.A.N., Thierens, D.: Expanding from discrete to continuous estimation of distribution algorithms: The IDEA. *Parallel Problem Solving from Nature PPSN VI*, pp. 760–776, Springer, 2000.