

RESEARCH ARTICLE | SEPTEMBER 06 2017

An attempt of reduction of optimization costs of complex industrial processes **FREE**

Łukasz Sztangret; Jan Kusiak



AIP Conf. Proc. 1872, 020020 (2017)

<https://doi.org/10.1063/1.4996677>



Boost Your Optics and Photonics Measurements

Lock-in Amplifier

Find out more

Boxcar Averager

An Attempt of Reduction of Optimization Costs of Complex Industrial Processes

Łukasz Sztangret^{1,a)} and Jan Kusiak^{1,b)}

¹AGH University of Science and Technology, al. Mickiewicza 30, 30-059 Kraków, Poland

^{a)}Corresponding author: szt@agh.edu.pl

^{b)}kusiak@agh.edu.pl

Abstract. Reduction of computational costs of optimization of real industrial processes is crucial, because the models of these processes are often complex and demand time consuming numerical computations. Iterative optimization procedures have to run the simulations many times and therefore the computational costs of the optimization may be unacceptable high. This is why a new optimization methods and strategies which need less simulation runs are searched. The paper is focused on the problem of reduction of computational costs of optimization procedure. The main goal is the presentation of developed by the Authors new, efficient Approximation Based Optimization (ABO) and Modified Approximation Based Optimization (MABO) methods which allow finding the global minimum in smaller number of objective function calls. Detailed algorithm of the MABO method as well as the results of tests using several benchmark functions are presented. The efficiency of MABO method was compared with heuristic methods and the results show that MABO method reduces the computational costs and improve the optimization accuracy.

INTRODUCTION

Models of the real industrial processes are often based on complex mathematical equations. The computer simulations, depending on their complexity and the available computational power, may last from few seconds to several hours. Therefore, the optimization of such processes is difficult (or sometimes impossible) due to the fact of large number of necessary objective function calls requiring time consuming simulations of such model. Another problem is multimodality of the optimized objective function. Classical optimization methods (direct search or gradient methods) return the first encountered minimum, which rarely is a global one. The use of heuristic methods significantly increases the probability of finding the global minimum, however due to the fact that most heuristic algorithms have to process the population of solutions, the number of simulations drastically increases. This is why the new optimization strategies, which allow to reduce computing time of optimization procedure, are searched. Two approaches dominate nowadays. The first one is focused on searching for faster, computationally more efficient models of optimized processes, the second one on developing new optimization methods, which allow finding the global minimum in smaller number of objective function calls. This research concerns the second trend. Within this paper the developed by the Authors the Approximation Based Optimization (ABO) and Modified Approximation Based Optimization (MABO) methods were presented [1, 2, 3]. Its main idea focus on the fact, that the optimal solution of considered objective function $f(\mathbf{x})$ is not searched directly, but indirectly. In each iteration, the analyzed objective function $f(\mathbf{x})$ is replaced by an approximation function $g(\mathbf{x})$ and the optimization procedure searches the optimal solution among solutions of a function $g(\mathbf{x})$. The acceleration of convergence has been achieved by introducing approximation weights in each point, which allow increasing the importance of the approximation points lying in the close neighborhood of actual minimum. Detailed algorithm of the developed method is presented in the paper, as well as the original algorithm dedicated for generation of the set of initial points, which is based on design of experiment theory. The algorithm responsible for changing values of weights during optimization procedure is also presented. The developed method was tested using several benchmark functions with different number of variables (from 1 to 5). The efficiency of MABO method was compared with most popular heuristic methods (genetic algorithm (GA), evolutionary algorithms (EA) and particle swarm optimization (PSO) method) [4, 5, 6, 7, 8, 9]. The results show that MABO method can significantly reduce the number of necessary objective function calls and gives better accuracy of solution.

APPROXIMATION BASED OPTIMIZATION METHOD

The Approximation Based Optimization (ABO) method is an iterative optimization technique [1, 2]. In each iteration, the analyzed objective function $f(\mathbf{x})$ is replaced by an approximation function $g(\mathbf{x})$ and the optimization procedure searches the optimal solution among minima of a function $g(\mathbf{x})$. Optimization procedure begins with establishing of the initial set $X^{(0)} \in D_x \subset \mathbb{R}^n$ consisting of feasible solutions. For each element $\mathbf{x} \in X^{(0)}$, $i = 1, 2, \dots, m$, the objective function value is computed and, as a result, a set $\{X^{(0)}, Y^{(0)} = f(X^{(0)})\}$ is created. Based on this set, coefficients of approximation function $g(\mathbf{x})$ are calculated by solving a system of linear equations expressing the approximation error ε in the approximation points:

$$\varepsilon = \sum_{i=1}^m (g(\mathbf{x}^i) - f(\mathbf{x}^i))^2. \quad (1)$$

Next, minimal solution of a function $g(\mathbf{x})$ is searched using any of the optimization techniques, and the found value \mathbf{x}^* is being added to a set $X^{(0)}$ resulting in a new data set $\{X^{(1)}, Y^{(1)}\}$. New approximation $g(\mathbf{x})$ of a function $f(\mathbf{x})$ is evaluated on the basis of that new data set, and the whole procedure is continued until stop condition is reached.

The efficiency of ABO method was evaluated using simple one-dimension test function:

$$f(x) = (\sin(x) - 2)^2 \sin(12x - 4), \quad (2)$$

where $x \in [-1, 1]$.

The polynomials of 2 to 9 degrees and a spline function were used as the approximation function $g(x)$. The test shows that only in case of spline function and high degree polynomial (9th degree) the ABO method was able to find the minimum of test function [3]. This results were the motivation to developing a modification of the ABO method to increase its efficiency.

MODIFIED APPROXIMATION BASED OPTIMIZATION METHOD

Achievement of the computing cost reduction and increasing the efficiency became possible after introducing some modification into ABO method. The following main modifications were implemented [3]:

1. introduction of weights w_i to all points in the set $X^{(0)}$ which describe their influence on coefficients of function $g(\mathbf{x})$,
2. development of the algorithm which generates the initial set $X^{(0)}$ based on design of experiment (DoE) theory,
3. choosing the approximation function $g(\mathbf{x})$,
4. development of the algorithm which generates weights of points of the set $X^{(0)}$,
5. development of the algorithm which changes the weights during the optimization procedure,
6. implementation of the second, local approximation of the function $g(\mathbf{x})$ based on points lying in the neighbourhood of current solution \mathbf{x}^* .

Some modifications were made and published in [10]. However, the method described in this paper was improved by the design of experiment theory.

Introduced weights, which correspond to the approximation errors at each point allow increasing of the importance of approximation points lying in the close neighborhood of the actual minimum \mathbf{x}^* . The values of weights w^i in each point depend on the values of the objective function at these points. The smaller is objective function value, the higher weight value should be. This modification yields the following form of equation (1):

$$\varepsilon = \sum_{i=1}^m w^i (g(\mathbf{x}^i) - f(\mathbf{x}^i))^2. \quad (3)$$

As the result of that modification, the function $g(\mathbf{x})$ is built based on the set $\{X^{(i)}, Y^{(i)}, W^{(i)}\}$.

Within the scope of this paper, the approximation function was limited to the square function:

$$g(\mathbf{x}) = \mathbf{x}^T \mathbf{A} \mathbf{x} + \mathbf{b}^T \mathbf{x} + c. \quad (4)$$

This choice was made due to two main reasons: computing square function coefficients is very easy even in high dimension cases and there is no need to use addition optimization procedure to find its minimum.

Selection of the initial data set $X^{(0)}$ is based on the design of experiment theory. The optimal plan of the experiment for one-dimension square function is composed of three points: two of them are the boundaries of the interval of interest and the third one was chosen in the middle. In case of higher dimension problems, the optimal plan is computed based on one-dimension plan [11].

Generation of initial data set is made in several stages (the number of stages is equal to e and its value can be changed). In each stage the initial data set is enlarged by adding new points generated according to DoE rules [12]. However, the coordinates of points are multiplied by the scaling factor λ^j and shifted by the vector μ^j . In the first stage the scaling factor and the shifting vector are equal $\lambda^1 = 1$ and $\mu^1 = \mathbf{0}$ respectively. In the subsequent stages the value of scaling factor is decreasing causing that new points are lying closer to each other. To compute the value of vector μ^j the coefficients of approximation function $g(\mathbf{x})$ are calculated based on all previously chosen points. The beginning of the vector is in a point $\mathbf{0}$ and its end is in the point in which the approximation function reaches the minimum. If the function $g(\mathbf{x})$ is concave then the vector $\mu^j = \mathbf{0}$. If, due to the shift, any point goes out of the domain D_x then it is forced back to the interior of the considered domain. Applying of this algorithm results in creation of the set $\{X^{(0)}, Y^{(0)}\}$. In the next step the weights of all points are calculated and added to the initial set. The method used to calculate the weights is very similar to the roulette wheel method known for genetic and evolutionary algorithms. In GA and EA the better individual gets higher probability of being selected for reproduction. In MABO method the better solution set gets higher value of weight.

Based on the set $\{X^{(i)}, Y^{(i)}, W^{(i)}\}$ the coefficients of $g(\mathbf{x})$ are determined. If the matrix \mathbf{A} (in equation 4) is positively defined, the minimum is calculated:

$$\mathbf{x}^* = -\frac{1}{2}\mathbf{A}^{-1}\mathbf{b}. \quad (5)$$

Next, if the point \mathbf{x}^* belongs to the domain D_x , its neighborhood of a radius r is investigated. If the number of points in that neighborhood is sufficient (i.e. the number of points is greater or equal to $0.5n^2 + 1.5n + 1$, where n is a dimension of \mathbf{x}), the local approximation is performed using another quadratic approximation function h . The minimum of function h becomes new, more accurate optimal solution \mathbf{x}^* . If the number of points is not sufficient the local approximation is not performed. Next, the values of the objective function $y^* = f(\mathbf{x}^*)$ and corresponding weight w^* are calculated. The weight w^* is computed as linear relationship between the maximal and minimal values of the weights $W^{(i)}$. The new point is added to the set $\{X^{(i+1)}, Y^{(i+1)}, W^{(i+1)}\} = \{X^{(i)} \cup \mathbf{x}^*, Y^{(i)} \cup y^*, W^{(i)} \cup w^*\}$ and the quantity of the set m is increased by one.

If the matrix \mathbf{A} is not positively defined or the point \mathbf{x}^* does not belong to the domain D_x then a point $\mathbf{x} \in X^{(0)}$ for which the objective function has the smallest value becomes a new minimal solution \mathbf{x}^* .

The weights w^j corresponding to all approximation points are modified in the next step. A first degree spline function $\psi(\eta)$ determined by three nodes U, V and Z is used in these modification (Figure 1a). The argument of this function is $\eta = \|\mathbf{x}^* - \mathbf{x}\|$, which is the distance of a given point from the current minimum.

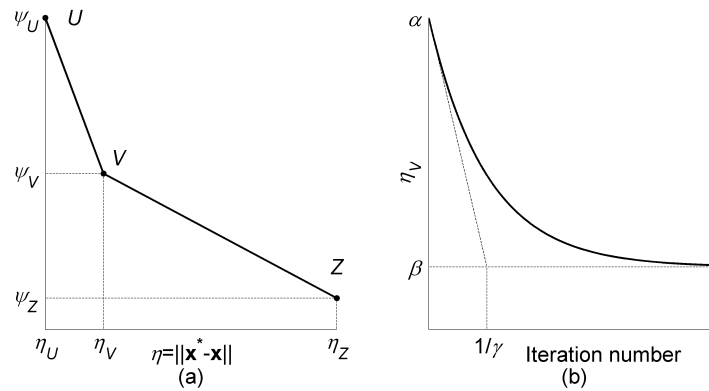


FIGURE 1. Spline function $\psi(\eta)$ controlling changes of the weights w^j (a) and the variation of an abscissa of a V node.

The coordinates of nodes U, V and Z determining a shape of a function $\psi(\eta)$ are:

$$\eta_U = 0$$

$$\begin{aligned}
\psi_U &= 1 \\
\eta_V &= (\alpha - \beta)e^{-\gamma i} \\
\psi_V &= 0.5 \\
\eta_Z &= \max\{\|\mathbf{x}^* - \mathbf{x}\|\} \\
\psi_Z &= 0.1
\end{aligned} \tag{6}$$

where: α and β are coefficients responsible for the initial and the final values of the abscissa of point V , while γ is responsible for the rate of changes and i is the iteration number.

Each value w^i is multiplied by the value of spline function $\psi(\eta)$. As the distance $\eta = \|\mathbf{x}^* - \mathbf{x}\|$ increases, the weight of the point becomes smaller. To improve the convergence, during the optimization procedure the abscissa of node V is decreasing, as it is shown in Figure 1b. Therefore, the weights of points lying far from actual minimum \mathbf{x}^* have even smaller impact on coefficients of approximation function $g(\mathbf{x})$.

The procedure ends when the maximum number of objective function calls N_{max} is reached or the obtained solution was satisfactory.

The MABO method applying to searching for the minimum of function (2) needed only 17 objective function calls to return optimal solution. The ABO method in case of using second order approximation function did not find the optimal solution in 1000 objective function calls. It shows that introduced modification increases the effectiveness of the method.

OPTIMIZATION RESULTS

Developed MABO method was validated by optimization of several two dimension benchmark functions:

- Ackley test function:

$$f(x_1, x_2) = -20e^{-0.2\sqrt{0.5(x_1^2+x_2^2)}} - e^{0.5(\cos(2\pi x_1)+\cos(2\pi x_2))} + 20 + e \tag{7}$$

where: $x_1, x_2 \in [-1, 5]$; value of optimal solution $y^{opt} = 0$,

- Michalewicz test function:

$$f(x_1, x_2) = -\sin(x_1) \left(\sin\left(\frac{x_1^2}{\pi}\right) \right)^{20} - \sin(x_2) \left(\sin\left(\frac{2x_2^2}{\pi}\right) \right)^{20} \tag{8}$$

where: $x_1, x_2 \in [-4, 4]$; value of optimal solution $y^{opt} = -1.8013$,

- Rastrigin test function:

$$f(x_1, x_2) = x_1^2 + x_2^2 - \cos(18x_1) - \cos(18x_2) + 2 \tag{9}$$

where: $x_1, x_2 \in [-1, 4]$; value of optimal solution $y^{opt} = 0$,

- Rosenbrock test function:

$$f(x_1, x_2) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2 \tag{10}$$

where: $x_1, x_2 \in [-2, 2]$; value of optimal solution $y^{opt} = 0$,

- Schwefel test function:

$$f(x_1, x_2) = -x_1 \sin(\sqrt{|x_1|}) - x_2 \sin(\sqrt{|x_2|}) \tag{11}$$

where: $x_1, x_2 \in [-500, 500]$; value of optimal solution $y^{opt} = -837.9658$.

The following stop conditions were considered: reaching the maximal number of objective function calls (N_{max}) or obtaining the optimization error (y^ε) lower or equal to assumed accuracy (ε). The optimization error was computed using the following formula:

$$y^\varepsilon = \frac{|y^{opt} - y^*|}{y^{opt}} \tag{12}$$

In case of Ackley, Rastrigin and Rosenbrock test functions the values of optimal solutions $y^{opt} = 0$, therefore the optimization error was assumed to be equal to the function value $y^\varepsilon = y^*$. The number of objective function calls N_{max} was set to 5000 and required accuracy $\varepsilon = 10^{-3}$.

The obtained results of the MABO method were compared with the results of chosen heuristic optimization methods: genetic algorithm, particle swarm optimization and evolutionary algorithms (strategies $(1 + 1)$, $(\mu + \lambda)$, (μ, λ)). The optimization procedure was performed 100 times for each of these methods and results presented in Figure 2 are the average values.

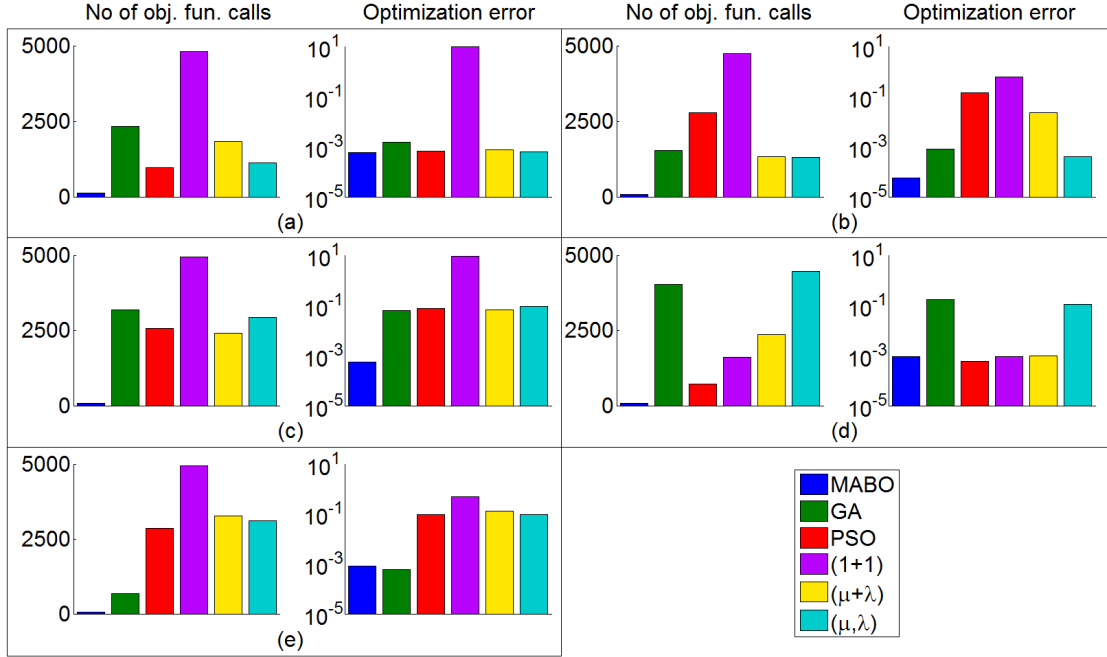


FIGURE 2. Results for Ackley (a), Michalewicz (b) Rastrigin (c), Rosenbrock (d) and Schwefel (e) test functions.

The performed tests were not limited to two-dimension functions. To evaluate the efficiency of MABO method during optimization of multi dimension problems a Levy test function was used:

$$f(\mathbf{x}) = \sin^2(\pi\tilde{x}_1) + \sum_{i=1}^{n-1} \left((\tilde{x}_i - 1)^2 (1 + 10 \sin^2(\pi\tilde{x}_i + 1)) \right) + (\tilde{x}_n - 1)^2 (1 + 10 \sin^2(2\pi\tilde{x}_n)) \quad (13)$$

where:

$$\tilde{x}_i = 1 + \frac{x_i - 1}{4} \quad (14)$$

and

$$\mathbf{x} \in [-5, 15]^n \quad (15)$$

The computations were made for $n = 1, \dots, 5$ variables. The value of optimal solution is equal to 0, therefore $y^\varepsilon = y^*$. The stop conditions were: $N_{max} = 5000$ and $\varepsilon = 10^{-3}$. The results are presented in Figure 3.

SUMMARY AND CONCLUSIONS

The main goal of the paper was presentation of elaborate Modified Approximation Based Optimization method. Developed method was validated in optimization of several benchmark functions, giving a similar optimal solution in much lower number of objective function calls comparing to the chosen heuristic algorithms. The reduction of

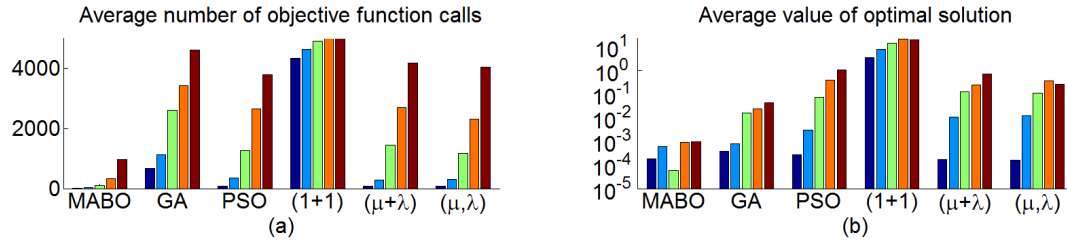


FIGURE 3. Results for Levy test functions of $n = 1, \dots, 5$ variables: average number of objective function calls (a) and average value of optimal solution (b).

objective function calls is very important in case of optimization of real industrial processes which are often described by complex models. It can be noticed, that in case of optimization of multi dimension function, the number of objective function calls used by MABO method grows exponentially with increasing of n . This is due to the optimal plan of experiment which is applied to generate the initial set of points. To reduce the quantity of this set the non-optimal plan, achieved by imposing the specific constraints on points can be used.

The obtained results confirm that the MABO method can significantly reduce the number of necessary objective function calls and hence to reduce the time of optimization process and can be useful in solving complex industrial optimization problems.

ACKNOWLEDGMENTS

Financial support of National Science Centre in Poland (NCN) is widely acknowledged – Grant No. DEC-2013/11/B/ST8/00352.

REFERENCES

- [1] J. Kusiak, *J. Mat. Proc. Techn.* **57**, 79–84 (1996).
- [2] J. Kusiak, A. Danielewska-Tulecka, and P. Oprocha, *Optimization. Chosen methods with applications* (PWN Wydawnictwo Naukowe, Warszawa, 2009 - in Polish).
- [3] L. Sztangret, “Reduction of computational costs of the metallurgical processes optimization,” PhD thesis - in Polish, AGH University of Science and Technology, Krakow, Poland 2014.
- [4] T. Back, *Evolutionary algorithms in theory and practice* (Oxford University Press, New York, 1996).
- [5] R. C. Eberhart and J. Kennedy, Proc. Conf. Sixth International Symposium on Micromachine and Human Science 39–43 (1995).
- [6] J. Holland, *Adaptation in natural and artificial systems* (MIT Press, 1975).
- [7] J. Kennedy and R. C. Eberhart, Proc. IEEE International Conference on Neural Networks 1942–1948 (1995).
- [8] H. Schwefel, *Evolution and optimum seeking* (Wiley, Chichester, 1995).
- [9] M. D. Vose, *The simple genetic algorithm: Foundations and Theory* (MIT Press, Cambridge, 1998).
- [10] L. Sztangret and J. Kusiak, *Lecture Notes in Artificial Intelligence* **2**, 600–607 (2012).
- [11] E. Rafajłowicz, *Algorytmy planowania eksperymentu: z implementacjami w środowisku Mathematica* (Akademicka Oficyna Wydawnicza PLJ, Warszawa, 1996 - in Polish).
- [12] R. H. Myers and D. C. Montgomery, *Response Surface Methodology: Process and Product Optimization Using Designed Experiments* (Wiley, New York, 1995).