Implementation and Testing of an Algorithm for Global Optimizations with Noise

Antanas Žilinskas and Darius Kučinskas

Abstract: *An implementation problem of an algorithm for one-dimensional global optimization in the presence of noise is considered. The approach is based on modelling the objective function as a standard Wiener process. The testing results of the implemented algorithm are presented using several well known test functions. The efficiency of the algorithm is evaluated for different levels of noise.*
Key words: *Software, Testing, Algorithms, Optimization.*

INTRODUCTION

The problem of global optimization of multimodal functions in the presence of noise is one of the most difficult in optimization theory. For the local optimization of functions with stochastic noise the stochastic approximation type methods are most popular [2]. Combination of a stochastic approximation algorithm with random multistart may be applied to global optimization of noisy functions [2]. However, the random multistart has many disadvantages as it is established in the case of global optimization without noise [7]. An alternative statistical approach is based on a statistical model of an objective function. Although implementation of algorithms based on statistical models are complicated, in the one-dimensional case special methods may be developed to reduce the computational complexity.

Several methods for global optimization in the presence of noise have been developed using the Wiener process for a model of the objective functions in [4], [9]. For an ALGOL code of a version of Wiener model based algorithm we refer to [8]. The theoretical investigation of such algorithms was started in the sixties but their implementation was prevented even in eighties by the limited computing power of the those days computers. The contemporary increase of computer power renewed the interest to global optimization in the presence of noise; see e.g. [1], [6].

In the present paper we consider the problems of algorithmic implementation of the method described in [1]. The implemented algorithm has been tested for the known test functions. The testing results show acceptable time of minimization of rather difficult test functions.

Description of the Algorithm

An objective function *f(x)* is to be minimized over the interval *[0,1]*; by rescaling we can treat the general interval. We sequentially choose points $x_1, x_2, \ldots$ at which to observe the function value depending on the results of the search. The observations are corrupted by random noise. Specifically, at the point x_i we observe $y_i = f(x_i) + \xi_i$, where the ξ_i are independent random variables with Gaussian distribution; assume their mean equal to *0* and their variance equal to σ^2. The Wiener process $W(x)$, $0 \le x \le 1$, is accepted as a statistical model of the objective function. After *n* observations we estimate the global minimum as

$$M_n = \min_{0 \le s \le 1} E(W(s)|F_n)$$

where $F_n = B\{x_i, y_i; i \le n\}$ represents the information available after the first *n* observations. Let us assume the observation points increasingly ordered. Then the conditional probability distribution of the random variable $W(x)$ with respect to F_n may be expressed via the multidimensional Gaussian density with covariance matrix

$$\Sigma = \begin{pmatrix} x_1+\sigma^2 & x_1 & x_1 & \cdots & x_1 \\ x_1 & x_2+\sigma^2 & x_2 & \cdots & x_2 \\ . & . & . & \cdots & . \\ . & . & . & \cdots & . \\ x_1 & x_2 & x_3 & \cdots & x_n+\sigma^2 \end{pmatrix},$$

since $\operatorname{cov}(W(t), W(\tau)) = \min(t, \tau)$.

To define a rational search strategy for minimum, a behaviour of the objective function should be forecasted. This can be done using conditional (given F_n) mean and variance of $W(x)$, $x_i \le x \le x_{i+1}$, which are defined by the following formulae

$$m(x|\cdot) = (y_1, \cdots, y_i, y_{i+1}, \cdots, y_n) \cdot S_N \cdot (x_1, \cdots, x_{i-1}, x_i, x \cdots x)^t, \tag{2.1}$$

$$s^2(x|\cdot) = x_i - (x_1, \cdots, x_i, x, \cdots, x) \cdot S_N \cdot (x_1, \cdots, x_i, x, \cdots, x)^t, \tag{2.2}$$

where $S_N = \Sigma^{-1}$. The complexity of a standard inversion of the covariance matrix Σ is $O(n^3)$. However, a lower complexity technique may be developed taking into account the special structure of Σ. The formulae given in [1] enable to develop an algorithm of the complexity $O(n^2)$ to invert the covariance matrix. The standard Wiener model $W(x)$ has a disadvantage with respect to practical applications of the corresponding optimization algorithm, since the equality $W(0)=0$ is valid with probability *1*. In practical situations no function value is known precisely as observations are made in the presence of noise. To cope with this problem, let us move the origin to $-\infty$. In the new coordinate system all differences x_{i+1}-x_i remain unchanged, only the first observation is placed at the origin: $x_1=0$.. Below we present the formulae based on the theory developed in [1]. The formula of conditional mean is given for an endpoint of a subinterval since conditional mean at the interior points of the subinterval $x_i<x<x_{i+1}$ can be obtained by means of linear interpolation. The formula for conditional variance at the point x, $d=x-x_i$ is given, since the function of conditional variance, as the quadratic function, can not be restored using the values at the ends of the subinterval. The formulae needed for the algorithm development are given by

$$m(x|\cdot) = \frac{r_i \sum_{j=1}^{i-1} y_j q_j + q_i \sum_{j=i}^{n} y_j r_j}{r_i \sum_{j=1}^{i-1} q_j + q_i \sum_{j=i}^{n} r_j} = \frac{\sum_{j=1}^{i-1} y_j \frac{q_j}{q_i} + \sum_{j=i}^{n} y_j \frac{r_j}{r_i}}{\sum_{j=1}^{i-1} \frac{q_j}{q_i} + \sum_{j=i}^{n} \frac{r_j}{r_i}}, \tag{2.3}$$

$$s(x|\cdot) = \sigma^2 \left(\frac{1}{\frac{Q_{t-1}}{q_i} + \frac{R_i}{r_i}} \cdot \frac{\Delta_{i+1}^2 - d^2}{\Delta_{i+1}^2} + \frac{1}{\frac{Q_i}{q_{i+1}} + \frac{R_{i+1}}{r_{i+1}}} \cdot \frac{d^2}{\Delta_{i+1}^2} \right) + \frac{d(\Delta_{i+1} - d)}{\Delta_{i+1}} \left(1 - 2 \frac{1}{\frac{Q_{i-1}}{q_i} \frac{r_i}{R_{i+1}} + \frac{R_i}{R_{i+1}}} \right), \tag{2.4}$$

where $\Delta_i = x_i - x_{i-1}$,

$$R_n = 1, r_n = 1, r_i = r_{i+1} + \frac{\Delta_{i+1} R_{i+1}}{\sigma^2}, R_i = R_{i+1} + r_i, i = n-1, \cdots, 1,$$

$$q_1 = 1, Q_1 = 1, q_i = q_{i-1} + \frac{\Delta_i Q_i}{\sigma^2}, Q_i = Q_{i-1} + q_i, i = 2, \cdots, n.$$

The P-algorithm performs the current *n+1*-th observation of the objective function value at the point where probability to find a better function value is maximal

$$x_{n+1} = \arg \max_{0 \le x \le 1} P\{W(x) < M_n - \varepsilon_n | F_n\}. \qquad (2.5)$$

The original problem of the minimization of *f(x)* over the interval $[0,1]$ is substituted with the minimization over the set of points $\chi_i = i/N, i=0,1,\ldots,N$. Correspondingly is modified implementation of P-algorithm (2.5). The auxiliary maximization problem over the continuous interval *[0, 1]* in (2.5) is replaced by the maximization problem over a finite set. The main advantage of such a modification is in avoiding of the necessity to maintain dynamically the ordered arrays of arbitrarily generated data x_i , y_i , where $x_i \le x_{i+1}$. The location of a points χ_i in the array is uniquelly defined, as well as the location of the corresponding function value. In case of the repeated observations at the same point the function values are averaged. The algorithm includes also the estimation of parameters of the model; we omit this question because of restricted extent of the paper.

Test functions

We will not discuss here the general methodology of testing optimization software; it is described e.g. in [5]. Some specific questions concerning global optimization are discussed in [3]. We use the test functions proposed in [7]. The optimization algorithm has access only to the noise corrupted function values $y_i = f(x_i) + \xi_i$, $1 \le i \le n$ where ξ_i are independent identically distributed random numbers with Gaussian density $N(0, \sigma^2)$. The used test functions are defined by the following formulae:

1. $f_1 = \sin x + \sin(10x/3) + \ln x - 0.84x + 3$, *where* $2.7 \le x \le 7.5$,
2. $f_2 = \sin x + \sin(2x/3)$, *where* $3.1 \le x \le 20.4$,
3. $f_3 = -\sum_{i=1}^{5} i \sin((i+1)x + i)$, *where* $-10 \le x \le 10$,
4. $f_4 = -\sum_{i=0}^{10} \frac{1}{(k_i(x - a_i))^2 + c_i}$, *where* $0 \le x \le 10$, and coefficients are presented in Table1

Table 1. Parameters of the function f_4

Test function	Coefficients	i=1	i=2	i=3	i=4	i=5
f_4	a_i	3.040	1.098	0.674	3.537	6.173
	k_i	2.983	2.378	2.439	1.168	2.406
	c_i	0.192	0.140	0.127	0.132	0.125
		i=6	*i=7*	*i=8*	*i=9*	*i=10*
	a_i	8.679	4.503	3.328	6.937	0.700
	k_i	1.236	2.868	1.378	2.348	2.268
	c_i	0.189	0.187	0.171	0.188	0.176

Table 2. Minimum values of the test functions

Test function	Minimum	Coordinates	Test function	Minimum	Coordinates
f_1	-1.6013075	5.1997784	f_3	-12.03122494	-6.7745760
f_2	-1.9059611	17.0391986	f_3	-12.03122494	-0.4913908
f_4	-14.5926520	0.6858609	f_3	-12.03122494	5.7917947

Experimental results

The goal of the experimental investigation was to test the applicability of the developed algorithm to find the global minimum of noisy multimodal functions, where the level of noise is comparable with the variance of the objective function. For the comparison we include the results obtained by the algorithm proposed in [8]. The number of iteration was chosen in the interval 100 – 1000, and the number of discretization points N in the interval 100 - 10000. Since the minimization result depends on the random noise, we present the averaged results.

Table 3. Minimization precision for the test function f_1

Functions	Number of iterations	Number of measurement points			
		100	3000	6000	10000
Algorithm of [8]	50	0.05	0.19	0.19	0.19
	75	0.21	0.01	0.1	0.04
	100	0.08	0.1	0.35	0.14
	1000	0.15	0.14	0.1	0.07
The developed algorithm	50	0.65	0.81	0.64	2.56
	75	0.17	0.25	0.68	0.64
	100	0.18	0.04	0.48	0.57
	1000	0.02	0.13	0.15	0.16

Table 4. Minimization precision for the test function f_2

Functions	Number of iterations	Number of measurement points			
		100	3000	6000	10000
Algorithm of [8]	50	0.35	0.33	0.33	0.33
	75	0.11	0.02	0.25	0.02
	100	0.02	0.07	0.21	0.01
	1000	0.01	0.03	0.17	0.13
The developed algorithm	50	0.11	0.12	0.11	0.13
	75	0.20	0.21	0.12	0.13
	100	0.04	0.04	0.14	0.08
	1000	0.07	0.24	0.12	0.16

Table 5. Minimization precision for the test function f_3

Functions	Number of iterations	Number of measurement points			
		100	3000	6000	10000
Algorithm of [8]	50	5.03	5.7	5.7	5.71
	75	1.17	2.93	3.06	2.88
	100	0.14	0.31	0.18	0.2
	1000	0.03	0.09	0.03	0.05
The developed algorithm	50	0.62	0.7	0.91	1.84
	75	0.39	0.3	0.41	0.56
	100	0.39	0.1	0.16	0.17
	1000	0.04	0.04	0.1	0.08

Table 6. Minimization precision for the test function f_4

Functions	Number of iterations	Number of measurement points			
		100	3000	6000	10000
Algorithm of [8]	50	3.05	3.3	3.4	3.14
	75	2.33	2.15	2.1	2.11
	100	2	1.76	1.78	1.94
	1000	0.11	1.61	1.65	1.67
The developed algorithm	50	9.5	8.1	8.3	8.4
	75	7.9	6.6	6.6	6.7
	100	2.6	6.4	6.4	6.5
	1000	0.8	0.3	0.24	0.26

The second aim of the experimental testing was to measure the computing time depending on the number of discretization points N. The computing time depends not only on the iteration number and the number of discretization points, but also it depends on objective function. In case a vicinity of a global minmizer is found fast, then the further search is less time consuming, since in such a case many observations are made repeatedly at the same points implying faster updating (2.3) and (2.4) than the updating performed in the case of the observation points scattered over all interval. 5000 iterations are performed to minimize all test functions. The dependence of computing time on the number of discretization points is presented in the Fig.1.

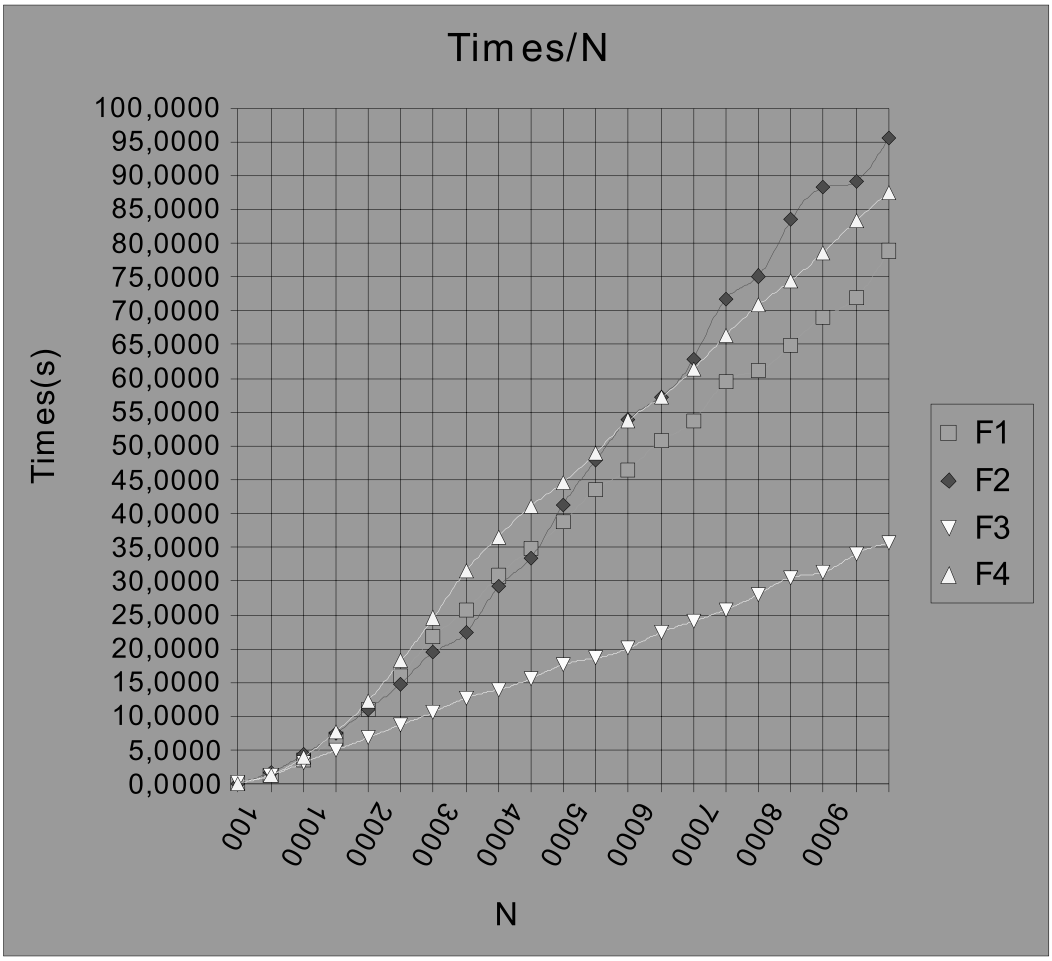

Fig.1 Time of 5000 iterations performed in minimization of six test functions

CONCLUSIONS AND FUTURE WORK

- The computing time of the developed algorithm increases approximately linearly with N , remaining acceptable for high precision discretization.
- The developed algorithm uses less iterations to achieve the same accuracy than the algorithm presented in [8].
- The further testing of the algorithm is planned to choose rationally some heuristic parameters of the algorithm. Final aim of the work is an extension of the algorithm to the multidimensional optimization using Peano type and other mappings of a one-dimensional interval to a multi-dimensional hypercube.

REFERENCES

[1] J. M. Calvin and A. Žilinskas, One dimensional optimization for observations with noise, Computers and Mathematics with Applications (submitted).

[2] H. Chen, Stochastic approximation and its applications, Kluwer Academic Publishers, 2002.

[3] C. Floudas and P. Pardalos, Handbook of test problems in local and global optimization, KAP, 1999.

[4] H. Kushner, A versatile stochastic model of a function of unknown and time-varying form, Journal of Mathematical Analysis and Applications 5 1962, 150/167.

[5] C. McGeoch, Experimental analysis of algorithms, Handbook of Global Optimization, Volume 2 (P. Pardalos and E. Romeijn, eds.), Kluwer: Dodrecht, 2002, pp. 489/514.

[6] R. Strongin and Y. Sergeyev, Global optimization with non-convex constraints, Kluwer, 2000.

[7] A. Törn and Žilinskas, Global optimization, Lecture Notes in Computer Science 350 (1989), 1/255.

[8] A. Žilinskas, Mimun-optimiztion of one-dimensional multimodal functions in the presence of noise, algoritmus 44, Aplikace Matematiky 25 (1980), 392/402.

[9] A. Žilinskas, Two algorithms for one-dimensional multimodal minimization, Mathematische Operationsforschung und Statistik, ser. Optimization 12 (1981), 53/63.

ABOUT THE AUTHORS

Antanas Zilinskas, Professor, Department of Informatics, Vytautas Magnus University, Lithuania, antanasz@ktl.mii.lt.

Darius Kucinskas, Ph.D. student, Department of Informatics, Vytautas Magnus University, Lithuania, Darius_Kucinskas@fc.vdu.lt.