UDC 519.67

TWO ALGORITHMS FOR GLOBAL OPTIMIZATION OF ONE-VARIABLE FUNCTIONS BASED ON THE SMALLEST ESTIMATE DISTANCES BETWEEN EXTREMES AND THEIR NUMBER

Kodnyanko V. A. – Dr. Sc., Professor, Polytechnic Institute, Siberian Federal University, Krasnoyarsk, Russia.

ABSTRACT

Contex. Making managerial decisions is often associated with solving one-dimensional global optimization problems. The most important property of global optimization methods is their speed, which is determined by the number of calls to the objective function in the optimization process.

Objective. Development of high-performance algorithms global for optimizing the function of one variable, based on conditions that allow you to bring the problem to a form that opens up the practical possibility of obtaining a solution with a given accuracy. Method. Two algorithms of conditional global optimization of a function of one variable are considered. The first is based on estimating the smallest distance between neighboring local extrema and allows you to find the global minimum of the goal function and, if necessary, all its local extrema. The second is suitable for finding the global minimum of a function if the number of local extrema in the uncertainty interval is known in advance. Both algorithms are based on segmentation methods of the initial uncertainty segment. The local extremum on a segment is determined by three or four points. An approach is proposed that, in most cases, allows localization of the extremum at three points, which provides savings in the calculation of digital filters, thereby contributing to an increase in the speed of the algorithm.

Results. The results of solving optimization problems and data on the effectiveness of the proposed algorithms are presented. A comparative analysis of the speed of the developed algorithms and well-known algorithms is carried out on the example of solving test problems used in world practice to assess the effectiveness of global optimization algorithms. Examples of the practical use of algorithms are given. The analysis of the data obtained showed that according to the number of calls to the objective function, the algorithms in the sequential computing mode work several times faster than modern high-speed algorithms with which they were compared.

Conclusions. The data presented indicate the efficiency and high speed of the proposed algorithms. Their speed will be even higher if the stated ideas of algorithmization are extended to parallel computations. This suggests that the proposed algorithms can find practical application in the global optimization of functions of the considered classes of problems.

KEYWORDS: function of one variable, local minimum of a function, global minimum of a function, global optimization, Brent's method, algorithm performance.

ABBREVIATIONS

TF – target function;

EPP – efficiency of a parallel process.

NOMENCLATURE

$f(x)$ is the objective function;

$G = [a, b]$ is segment of uncertainty;

ε is the accuracy of determining the global extremum;

d_{min} is the smallest distance between the extremums of the CF;

n is the number of segments on the segment G;

h is the length of the segment;

k_{min} is the smallest number of local extrema on the segment G;

k is the number of TF calls in optimization process.

INTRODUCTION

Making managerial decisions is often associated with solving one-dimensional global optimization problems. Theoretically, such tasks are unsolvable. Therefore, optimization is usually carried out in a given area. However, even in this case, the problem of conditional global optimization is too complex, because it often requires unacceptably large costs of computing resources and / or calculation time. For this reason, conditions are usually imposed on the target function in order to bring the problem to a form that opens up the practical possibility of obtaining its solution with a given accuracy [1 – 5].

1 PROBLEM STATEMENT

The most important property of global optimization methods is their speed. It is believed that the less computation of a minimized function a method requires to achieve a result, the more efficient it is and the higher its speed. High performance is necessary in two cases: when calculating a single value of a function requires a significant amount of computer time (for example, in some problems of structural mechanics it often takes several hours for a computer to calculate the objective function [24]) and when mlti-criteria optimization of the model of an object or phenomenon is carried out, which often also requires a significant investment of time.

In this paper, we consider two such deterministic algorithms that are based on certain constraints and conditions. Examples of the use of algorithms for solving practical problems are given.

Both algorithms solve the problem of finding the global minimum of the function $f(x)$ of one variable on the uncertainty interval $G = [a, b]$ up to ε. The condition for the operability of the algorithms is the absence on the segment G of sections of the constancy of TF. This means that for $\forall\ x \in G$ the equality

$$f(x-\varepsilon) = f(x) = f(x+\varepsilon). \qquad (1)$$

should not be satisfied.

DOI 10.15588/1607-3274-2020-2-4

2 REVIEW OF THE LITERATURE

Numerous approaches to global optimization are known, but exact methods are most widely used in practice, among which are methods based on a priori information, for example, data on the rate of change of a function over a given interval based on estimates using the Lipschitz constant. Currently, the approach to the development of algorithms for solving problems by Lipschitz global optimization methods is considered the most promising [6, 8, 10]. The fact that this direction in the field of the development of global optimization methods is currently being preferred is indicated by the achievements of leading scientists noted by numerous awards [7–10].

Prospects for the development of optimization methods are also associated with the use of parallel computing. Such an approach to the construction of methods can significantly reduce the optimization time through parallel computing [12–13].

In the practice of global optimization, methods are used that are usually divided into stochastic and deterministic [14, 15].

Stochastic algorithms are usually used to solve multidimensional problems, since they investigate functions at random points and perform additional calculations that lead to the goal, if such points turn out to be promising when some criterion is fulfilled. However, such algorithms do not guarantee success.

When solving one-dimensional problems, stochastic algorithms have no advantages over deterministic ones, because the latter allow an exhaustive search in the field of uncertainty and, therefore, have better prospects in comparison with the former, and also favor parallel computations. From the point of view of speed, preference is also given to deterministic algorithms [23].

3 MATERIALS AND METHODS

The first algorithm is based on the estimation of neighboring extrema (algorithm A1).

Consider the class of global optimization problems for which the estimate $d \le d_{min}$, where $d_{min} > \varepsilon$ is known.

Divide the segment G into $n = 1+[(b–a)/(d–\varepsilon)]$ equal segments. Obviously, the TF is strictly unimodal [17, 22] on the segment $G_i = [x_i, x_{i+1}]$, where $x_j = a+jh$, $j = 0, 1, \ldots, n$, $h = (b–a)/n$, i. e., each of them may contain either one maximum, or one minimum, or a OF on it is monotonous.

We define a structure of the type $t = t\ (x, i, p_1, p_2)$, where x is a number, i is a logical variable, $p_j = (p_j.x, p_j.y)$ are points, $j = 1, 2$. For the ends the current segment, we introduce the variables w, v of this type.

Put $x = a$, $w.x = x$, $w.p_1.x = x – \varepsilon$ and calculate $w.p_1.y = f(w.p_1.x)$. We take $w.i = 0$. This means that $w.p_2$ is not yet defined. We set $v.x = x + h$, $v.i = 0$ and calculate $v.p_1.y$.

If $w.p_1.y < v.p_1.y$, then we put $w.p_2.x = x + \varepsilon$, $w.i = 1$ and calculate $w.p_2.y = f(w.p_2.x)$, otherwise we similarly define v. If $w.p_1.y = w.p_2.y$, then put $w.p_2.x = x$ and calculate $w.p_2.y$. Moreover, by virtue of (1), $w.p_1.y \ne w.p_2.y$ is guaranteed. The same goes for v.

Now two points $w.p_1$, $v.p_1$ and one of the points $w.p_2$ or $v.p_2$ are calculated at the ends of the segment. Sort this sequence by x. If the midpoint has a minimum in y, then this segment contains the local minimum of the TF, if the maximum, then the local maximum.

If the given sequence is monotonic in y, then we define w or v for which $i = 0$. Now four points are calculated at the ends of the segment of TF – two at each edge. Using their gradient $w.p_2.y – w.p_1.y$ and $v.p_2.y – v.p_1.y$, we can determine the behavior of the function at the ends of the segment and, on this basis, conclude that there are local extrema provided that the sequence $w.p_1.y$, $w.p_2.y$, $v.p_1.y$, $v.p_2.y$ is non-monotonous.

Thus, a local extremum on a segment can be determined by three or four points. If the localization of the extremum can be determined by three points, then this gives savings in one calculation of TF.

Following a similar scheme, we will scan the remaining segments of the segment G.

Now it remains only to apply the method of local optimization to each segment that contains an extremum and successively find all the minima, and if necessary all the maxima.

The second algorithm is based on a known number of local minima (algorithm A2).

Consider the class of problems for which k_{min} is known.

Create an array of $\{w_i\}$ segment border structures. We take $w_0.x = a$ and $w_1.x = b$ and calculate w_0 and w_1 according to the above method.

We set $n = 1$, $h = b – a$, and in the general case for $n \ge 1$, starting from $n = 1$, we arrange a passage through the points $x_i = a + (i – 0{,}5)\ h$, $i = 1, 2,\ldots, n$ of an array w_i. Sort array w by field x in ascending order.

As a result, adding an element to the array w forms two new segments. We make a passage through all segments and find the number of lows in accordance with the above methodology. If the number of such segments coincides with k_{min}, then we stop the process of dividing the segment into segments, perform local minimization of TF in segments containing minima, and finish the work. Otherwise, we continue to search for segments until the condition $i = n$ is fulfilled. After completing such a cycle, increase n and halve h and make a new cycle pass with counter i. If it is not possible to find the required number of minima, then we continue the process until $h > \varepsilon$. If this condition is not met, the search process is completed. In this case, the last recorded minimum will be the solution to the problem.

Note that if the specified number of minima k_{min} does not correspond to the number k_m of minima of TF on the segment G, then for $k_{min} < k_m$, the solution found may turn out to be erroneous. If $k_{min} > k_m$, then the search for segments will take too long, even longer than using the simple enumeration method. In this case, it is convenient to introduce a new input parameter n_f, which imposes a limitation on the number of calls to TF. When going beyond its limits, the algorithm finds a solution to the

© Kodnyanko V. A., 2020
DOI 10.15588/1607-3274-2020-2-4

problem based on the use of data accumulated during the optimization process. With a sufficiently large value of the parameter n_f, as a rule, the algorithm gives the correct result, but in the general case it is not guaranteed that the solution obtained will turn out to be exact.

4 EXPERIMENTS

As an example of application of algorithm A1, we consider the results of global function optimization

$$f(x) = \sin\left[38(x-0.4)^2\right] + (x-0.6)^2 + 1.2$$

obtained by algorithm A1 on the interval $G = [a, b]$, $a = 0$, $b = 1$. Hereinafter, the accuracy $\varepsilon = (b - a)\ 10^{-4}$. The function graph is shown in Figure 1.

Accepted $d = 0.082$. To search for local extremes, the Brent method [18 – 21] was used, adapted to solve the problem taking into account data on the calculated TF values obtained by scanning the ends of the segments. As shown below, such an adaptation of the method accelerates the work of algorithms.

Figure 2 shows the application window, which shows the initial data and the results of sequential global optimization of the digital filters. Abscissa of the global minimum of TF $x = 0.7517$.

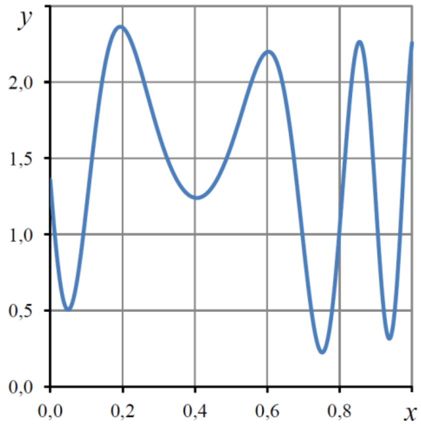
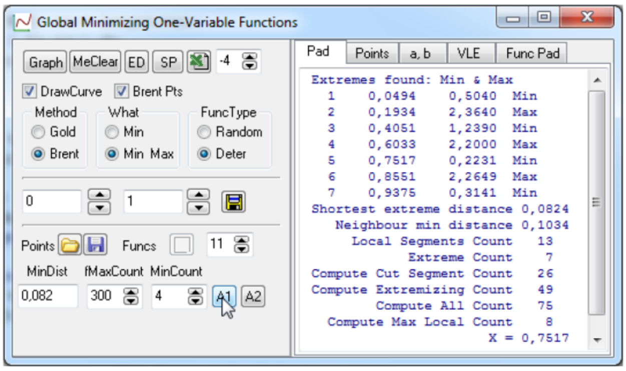


Figure 1 – Function graph

On the segment, 7 local extrema of the function were found, of which 4 turned out to be minima and 3 maxima. The calculated smallest distance between adjacent extrema is $d_{\min} = 0.0824$. The number of local segments is $n = 13$. To search for the behavior of the function on the segments, $k_1 = 26$ calls to the function were required, and to search for local extrema by the Brent method $k_2 = 49$ calls to it.

Thus, in the optimization process, it was necessary to calculate the digital filter $k = k_1 + k_2 = 75$ times. The minimum distance between adjacent extrema $d = 0.1034$.

When searching for local extremes, the slowest process required 8 calculations of TF. This means that if the calculations were not carried out sequentially, but in parallel with a sufficient number of processors, the efficiency of the search for local extrema would increase by 49/8 ≈ 8 times. In this case, the separation efficiency of segments would increase by 26/2 ≈ 13 times, since identification of one segment requires no more than two calculations of digital filters. Thus, in comparison with a sequential process, the efficiency of a parallel process in solving this problem will be no less than (26 + 49) / (8 + 2) = 7.5 times higher. If there are a sufficient number of processors, a similar algorithm will have linear acceleration [22] and this problem will be solved faster. Note that such an algorithm can be optimized to the superlinear acceleration index, since the detected monotonicity segments of the digital filters do not require local optimization and, therefore, the problem can be solved with practically the same acceleration on a smaller number of processors. This circumstance is essential for parallel multidimensional global optimization, requiring a significant number of processors [23].



-4

☑ DrawCurve ☑ Brent Pts

Method: ○ Gold ● Brent

What: ○ Min ● Min Max

FuncType: ○ Random ● Deter

0 1

Points Funcs □ 11

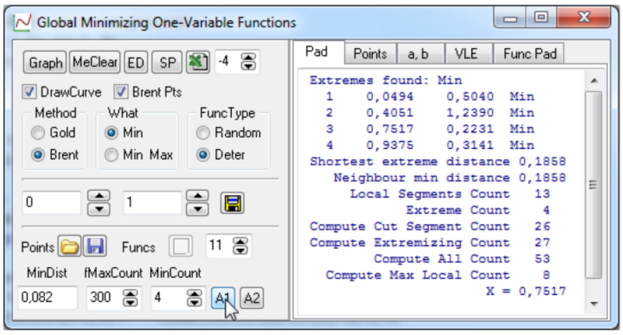
MinDist 0,082 fMaxCount 300 MinCount 4

```
Extremes found: Min & Max
  1     0,0494     0,5040  Min
  2     0,1934     2,3640  Max
  3     0,4051     1,2390  Min
  4     0,6033     2,2000  Max
  5     0,7517     0,2231  Min
  6     0,8551     2,2649  Max
  7     0,9375     0,3141  Min
Shortest extreme distance 0,0824
   Neighbour min distance 0,1034
     Local Segments Count   13
            Extreme Count    7
Compute Cut Segment Count   26
Compute Extremizing Count   49
        Compute All Count   75
  Compute Max Local Count    8
                X = 0,7517
```

Figure 2 – Application window

For comparison, we note that if the calculations were carried out using the direct search method (simple search with step ε) [5], then 10,000 calls to TF would be required.

Figure 3 shows the search results for the global minimum of TF. In this case, the search for local maxima is not performed.



-4

☑ DrawCurve ☑ Brent Pts

Method: ○ Gold ● Brent

What: ● Min ○ Min Max

FuncType: ○ Random ● Deter

0 1

Points Funcs □ 11

MinDist 0,082 fMaxCount 300 MinCount 4

```
Extremes found: Min
  1     0,0494     0,5040  Min
  2     0,4051     1,2390  Min
  3     0,7517     0,2231  Min
  4     0,9375     0,3141  Min
Shortest extreme distance 0,1858
   Neighbour min distance 0,1858
     Local Segments Count   13
            Extreme Count    4
Compute Cut Segment Count   26
Compute Extremizing Count   27
        Compute All Count   53
  Compute Max Local Count    8
                X = 0,7517
```

Figure 3 – Application window

As can be seen from the Figure 3 data, the number of calls to TF decreased to 53. At the same time, 26 calls to the TF were required to separate the segments; when searching for local minima, the TF was calculated 27 times.

© Kodnyanko V. A., 2020
DOI 10.15588/1607-3274-2020-2-4

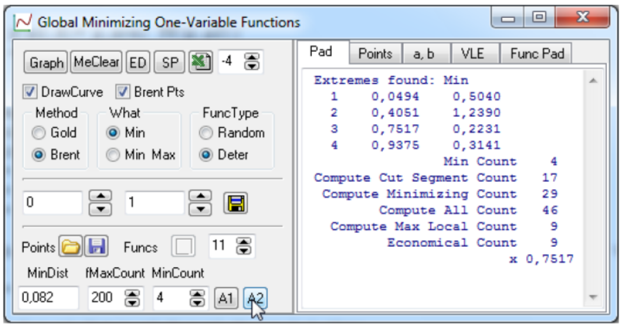
Table 1 shows the indicators of the influence of the estimate d on the speed of the algorithm, the value of which is inversely proportional to the number of calls to the objective function. Parameters *D* and *K* are, respectively, the ratios of *d* and *k* to their values in the line No. 1 of Table 1.

Table 1 - Indicators of the impact of the assessment *d* on algorithm performance

№ п/п	*d*	*N*	k_1	k_2	*k*	*D*	*K*
1	0.08	21	27	26	53	1	1.0
2	0.008	201	254	21	275	10	5.2
3	0.0008	2001	2504	17	2521	100	47.6

The data in Table 1 indicate that with a decrease in the estimate of *d* by 10 times, the speed of the algorithm decreases by about 5 times, i. e., solving the problem requires about 5 times more calculations of TF.

Figure 4 shows the results of global optimization for the above function according to algorithm A2 for the case when the number of minima $k_{\min} = 4$ is set correctly.



-4

☑ DrawCurve ☑ Brent Pts

Method: ○ Gold ● Brent

What: ● Min ○ Min Max

FuncType: ○ Random ● Deter

0 1

11

MinDist 0,082 fMaxCount 200 MinCount 4

```
Extremes found: Min
   1     0,0494    0,5040
   2     0,4051    1,2390
   3     0,7517    0,2231
   4     0,9375    0,3141
                 Min Count      4
 Compute Cut Segment Count     17
  Compute Minimizing Count     29
         Compute All Count     46
    Compute Max Local Count     9
       Economical Count         9
                           x 0,7517
```

Figure 4 – Application window

As follows from a comparison of the above results, the algorithms A1 and A2 have approximately the same speed. Indeed, if A1 performed local minimization in 53 DF calculations, then A2 requires 46 accesses to it. Interestingly, for method A2, the savings in segment detection are significantly higher and amount to 9 TF calculations, which allowed to increase the algorithm performance by about 20%.

If we optimize the function for $k_{\min} = 1 < 4$, when the number of minima is set incorrectly, then we get an erroneous result $x = 0.4051$.

If we set $k_{\min} > 4$, then for $n_f \geq 46$ the algorithm will find the right solution and indicate the exact number of detected minima.

With global optimization of functions with a sufficiently large number of extrema, the A1 and A2 algorithms more often showed close performance, however, examples of solving problems when the latter worked faster are not uncommon.

Figure 5 shows an example of such a task. The function has 34 local extrema. The smallest distance between its extrema is dmin = 0.0284.

Algorithm A1 required 533 TF calculations to determine all extrema. At the same time, 326 computations of TF and 207 computations for local optimization were used to separate the segments. When searching for seventeen local minima, the TF was calculated 425 times, 326 calls to the DF were used to separate the segments, and 99 to local minimization.

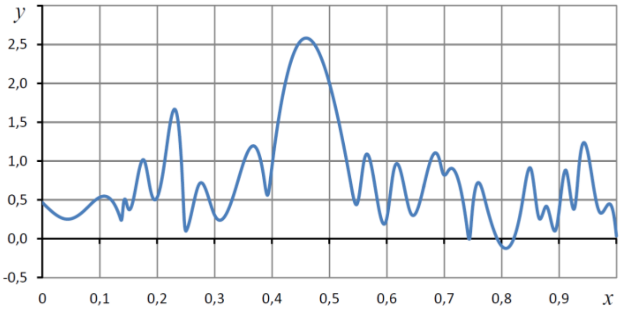


Figure 5 – Function graph

Using Algorithm A2, the problem was solved in 266 DF calculations, i.e., 1.6 times faster. At the same time, 148 calculations of the TF went to the separation of segments and 118 to the local minimization of TF.

Note that there is the possibility of accelerating the operation of algorithm A1. It consists in the fact that if the estimate $d_n \geq d$ of the smallest distance between extrema adjacent to the global minimum is known, then for $d_n > d > d_{min}$ the problem can be solved faster. For the problem under consideration, the smallest distance between extrema adjacent to the global minimum is $d_n = 0.0419$. For $d = d_n$, to find a global minimum, A1 algorithm needed only 124 TF calculations, i. e., the problem was solved 425/124 ≈ 3.4 times faster. It is guaranteed that the task of finding the global minimum will be solved by algorithm A1 without errors.

5 RESULTS

We present the results of a comparative analysis of the performance of the algorithms using the example of solving several test problems [16], shown in Table 2. Let us compare the operation of the algorithms A1 and A2 with the algorithms described in the fundamental work of Ya. D. Sergeev and D. E. Kvasov [8], research which are supported by grants from the Russian Foundation for Basic Research, the Council on Grants of the President of the Russian Federation, and the Italian Foundation for Basic Research.

Table 2 – Test problems

No	Function	Segment
1	$\sin x + \sin \frac{10x}{3}$	[2.7, 7.5]
2	$-\left(16x^2 - 24x + 5\right)\exp\left(-x\right)$	[1.9, 3.9]
3	$-x \sin x$	[0, 10]
4	$x^6 - 15x^4 + 27x^2 + 250$	[–4, 4]
5	$2\left(x-3\right)^2 + \exp\left(0.5x^2\right)$	[–3, 3]
6	$\sin^3 x + \cos^3 x$	[0, 6.28]

© Kodnyanko V. A., 2020
DOI 10.15588/1607-3274-2020-2-4

Table 3 contains data on the number of calls by algorithms to the digital filter. The “NA2”, “NA”, “I-NG” columns provide data on the number of calls to the digital filters by the best third-party algorithms, and the “A1”, “A2” columns give similar indicators of the proposed algorithms. The “E” column indicates the number of local extrema of the functions corresponding to table 2. The “(A1 + A2) / 2” column shows the average values for the A1 and A2 algorithms. The last column shows the indicators of the best third-party I-NG algorithm reduced to the average of the proposed methods.

The data contained in table 3 indicate that according to the number of calls to the digital algorithms in sequential computing mode, they work several times faster than the algorithms with which the comparison was made.

The data presented in the “A1” column allows us to evaluate the effectiveness of the adapted Brent method in comparison with the brackets of the original Brent method. It can be seen that the adapted method gives almost the same results (problem No. 6), or allows to increase the speed of the algorithm. So, for problem No. 3, the adapted algorithm works 35% faster (17 TF calculations versus 23).

With an increase in the number of extrema, the efficiency of the proposed algorithms decreases, but remains very high. So, even the worst of the indicators of the average relative speed of the algorithms A1 and A2, when solving problem No. 1, is 5 times higher than the characteristics of the best third-party algorithms. The calculations showed that when parallelizing the calculations, the comparative speed of algorithm A1 in solving this problem can increase by 14 times.

6 DISCUSSION

We give an example of the practical use of algorithm A1.

Consider the problem of determining the global minimum of a function given in the form of a table of experimental data.

Using them, we construct a cubic B-spline [24, 25]. Its graph is shown in Fig. 6.

Since there can be no more than one minimum on each segment of the cubic spline, the smallest distance between adjacent minima can be estimated by the smallest distance between neighboring abscissas x_i, which in this case is d = 0.01. The solution x = 0.287 of the problem was obtained for 62 calls of the A1 algorithm to the spline.

As an example of the practical use of algorithm A2, we consider the solution to the problem of determining the longest aperiodic component of the transient process of a linear dynamic system [26].

The characteristic polynomial of the dynamical system is selected, whose order is 42. It is known that aperiodic components are characterized by the real roots of the polynomial [24]. Therefore, to solve the problem, it is necessary to find its largest real root.

Having compiled a sequence of Sturm polynomials [27, 28], we established the number of real roots of the polynomial k = 5 for 42 calls to the polynomial. Applying algorithm A2 to the function of the square of the polynomial whose root corresponds to its global minima, we found the required root for 37 calls to the polynomial.

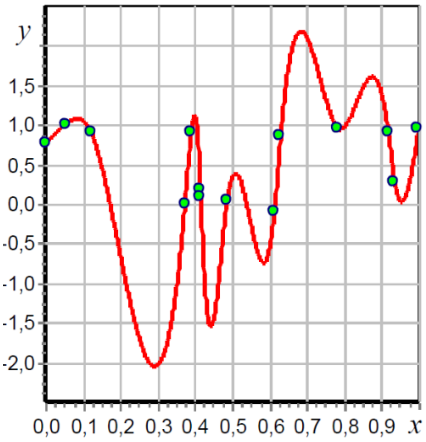


Figure 6 – Spline graph

Table 3 – Indicators of comparative testing of algorithms

№ п/п	Y. D. Sergeev, D. E. Kvasov [8, C. 86–88]			Proposed algorithms				(I–NG) / (A1+A2)/2
	NA2	NA	I–NG	E	A1	A2	(A1+A2)/2	
1	308	155	135	5	22 (25)	31	27	5
2	923	413	379	1	12 (14)	12	12	32
3	416	203	157	4	17 (23)	22	20	8
4	1412	549	470	5	53 (60)	76	65	7
5	992	497	557	1	16 (20)	20	18	31
6	746	327	271	7	25 (31)	37	31	9

Table 4 – Test problems

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13
x_i	0.00	0.05	0.12	0.37	0.39	0.41	0.42	0.48	0.61	0.63	0.78	0.91	0.93	1.00
y_i	0.76	1.00	0.90	0.00	0.91	0.19	0.09	0.05	0.10	0.86	0.95	0.90	0.29	0.95

DOI 10.15588/1607-3274-2020-2-4

The same root was found in the traditional way using the Bairstow-Hitchcock method [29] to search for all the roots of the polynomial with the subsequent separation of the required root. This decision was found for 254 calls to the polynomial. The use of the Sturm sequence and A2 algorithm instead of the traditional method of searching for polynomial roots allowed us to reduce the number of calls to the polynomial by 254 / (42 + 37) = 3.2 times, thereby significantly increasing the performance of the system dynamics quality analysis algorithm.

CONCLUSIONS

Two algorithms for global optimization of one variable functions are proposed. The advantage of algorithms is their high speed, which is confirmed by examples of solving numerous optimization problems, as well as by comparison with the best algorithms for global minimization of functions. The practical significance of the results lies in the ability of algorithms to find a solution faster than known algorithms, which is confirmed by an example of solving two problems of practical importance.

The presented data indicate the efficiency and speed of the algorithms A1 and A2. Their speed will be even higher if the declared ideas of algorithmization extend to parallel computing.

This suggests that the proposed algorithms can find practical application in global optimization of functions of the considered classes of problems.

Prospects for further research are the creation of global minimization algorithms for functions of many variables.

ACKNOWLEDGEMENTS

The work was carried out within the framework of the scientific research budget themes «Methods of modeling and optimizing of quality control information systems on the basis of intellectual technologies» at the Department of Standardization, Metrology and Quality Management of the Polytechnic Institute of the Siberian Federal University.

REFERENCES

1. Bryson A. E., Ho Y. Applied Optimal Control: Optimization, Estimation and Control. Hemisphere Publishing Corporation, Washington DC, 1975, 477 p.
2. Tan Y. Nešić D., Mareels M. Y., Astolfi A. Automatica. *On global extremum seeking in the presence of local extrema,* 2009, Vol. 45 (1), pp. 245–251. https://doi.org/10.1016/j.automatica.2008.06.010
3. Bizon N. Energy optimization of fuel cell system by using global extremum seeking algorithm, *Applied Energy*. – 2017, Vol. 206, pp. 458–474. https://doi.org/10.1016/j.apenergy.2017.08.097
4. Bertsekas D. P., Tsitsiklis J. N. Parallel and Distributed Computation: Numerical Methods. New Jersey, Prentice Hall, 1989, 94 p.
5. Bomze I. M., Csendes T., Horst R., Pardalos P. M. Developments in Global Optimization. London, Kluwer, 1997, 348 p.
6. Kvasov D. E., Sergeev Ya. D. Methods of Lipschitz global optimization in control problems, *Automation and telemechanics*, 2013, No. 9, pp. 3–19.
7. Sergeev Ya. D., Kvasov D. E. Adaptive diagonal curves and their software implementation, *Bulletin of Nizhny Novgorod State University: Mathematical Modeling and Optimal Control*, 2001, Vol. 2, No. 24, pp. 300–317.
8. Sergeev Ya. D., Kvasov D. E. Diagonal methods of global optimization. Moscow, FIZMATLIT, 2008, 276 p.
9. Pizzuti C., Sergeyev Ya. D. Comparison of two partitionstrategies in diagonal global optimization algorithms: Technical report 9. – Institute of Systems and Informatics. Italy, Rende, 2001, 16 p.
10. Dennis J. E. Jr., Torczon V. Direct Search Methods on Parallel Machines, *SIAM Journal Optimization*, 1991, pp. 448–474.
11. Nemhauster G. L., Pruul E. A., Rushmeier R. A. Branch-and-bound and Parallel Computation: a Historical Note, Operations Research Letters, 1988. No 7, pp. 65–69.
12. Orlyanskaya I. V., Perunova Y. N., Zavriev S. K. An Implementation of the Parallel Control Algorithm for Global Optimization, *Proceedings of the 3rd Moscow International Conference on the Research of Operations,* 2001, pp. 91–92.
13. Ali M., Törn A., Viitanen S. Stochastic Global Optimization: Problem, Classes and Solution Techniques, *Journal of Global Optimization*, 1999, No 14, pp. 437–447.
14. Moccus J. Application of Bayesian Approach to Numerical Methods of Global and Stochastic Optimization, *Journal Global Optimization,* 1994,Vol. 4, No. 4, pp. 347–356.
15. Hansen P., Jaumard B. Lipschitz optimization, Handbook of global optimization, 1995, Vol. 1, pp. 407–493.
16. Kodnyanko V. A. Optimization of unimodal functions by the parabolic predictor method, *Bulletin of the Astrakhan State Technical University. Series: Management, Computing and Informatics*, 2018, No. 3, pp. 101–108. DOI: 10.24143 / 2072-9502-2018-3-101-108
17. Brent R. P. Algorithms for Minimization Without Derivatives. Dover, 2002, 195 p.
18. Brent R. P. Errata for Algorithms for Minimization without Derivatives. – Prentice-Hall edition, 1973. Available from: https://maths-people.anu.edu.au/~brent/pub/pub011_errata.html
19. Burkardt J. V. Brent algorithms for minimization without derivatives, 2008. Available from: http://people.sc.fsu.edu/~jburkardt/f77_src/brent/brent.html
20. Rheinboldt W. C., Burkardt J. V. Algorithm 596: a program for a locally parameterized, 2008. Available from: http://people.sc.fsu.edu/~jburkardt/f77_src/brent/brent.f
21. Rao, S. S. Engineering optimization: theory and practice. John Wiley & Sons, Inc., Hoboken, New Jersey, 2009, 813 p.
22. Amdahl G. The Validity of Single Processor Approach to Achieving Large Scale Computing Capabilities, *Spring Joint Computer Conference*, 1967, Vol. 30, pp. 483–485.
23. Nocedal J. Theory of algorithms for unconstrained optimization. Acta Numerica, Serles A, Cambridge University Press, Cambridge, 1991, pp. 199–242.
24. Birkhoff S., de Boor C. R. Piecewise polynomial interpolation and approx-imation, *Garabedian, General Motors Symposium, 1964, Elsevier*, New York and Amsterdam, 1965, pp. 164–190.
25. Epps B. P., Truscott T. T., Techet A. H. Evaluating derivatives of experimental data using smoothing splines,

© Kodnyanko V. A., 2020
DOI 10.15588/1607-3274-2020-2-4

3rd Mathematical Methods in Engineering International Symposium MME'10, Coimbra, Portugal, 2010, pp. 21–24.

26. Kodnyanko V., Kurzakov A. Quality of Dynamics of Gas-static Thrust Bearing with Movable Carrying Circle on Elastic Suspension / V. Kodnyanko, A. Kurzakov, *Tribology in Industry,* 2019, Vol. 41, No. 2, pp. 237–241. DOI: 10.24874/ti.2019.41.02.09
27. Kurosh A. Higher Algebra. Moscow, Mir Publishers, 1984, P. 432.
28. Camargo Brunetto M. A. O., Claudio D. M., Trevisan V. An algebraic algorithm to isolate complex polynomial zeros using Sturm sequences, *Computers & Mathematics with Applications,* 2000, Vol. 39 (3–4), pp. 95–105. https://doi.org/10.1016/S0898-1221(99)00336-3
29. Vandergraft J. S. Certification of Algorithm 3: Solution of polynomial equations by Bairstow-Hitchcock method, *Communications of the ACM*, 1961, Vol. 4, No 2, pp. 105–117.

Received 28.01.2020.
Accepted 01.03.2020.

УДК 519.67

ДВА АЛГОРИТМА ГЛОБАЛЬНОЇ ОПТИМІЗАЦІЇ ФУНКЦІЙ ОДНІЄЇ ЗМІННОЇ, ЗАСНОВАНІ НА ВІДСТАНІ МІЖ ЕКСТРЕМУМАМИ ЇХНЬОЇ КІЛЬКОСТІ

Коднянко В. А. – д-р техн. наук, професор кафедри стандартизації, метрології та управління якістю. Політехнічний інститут Сибірського федерального університету, Красноярськ, Росія.

АНОТАЦІЯ

Актуальність. Прийняття управлінських рішень часто пов'язане з вирішенням завдань одновимірної глобальної оптимізації. Найважливішим властивістю методів глобальної оптимізації є їхня швидкодія, яке визначається кількістю звернень до цільової функції (ЦФ) в процесі оптимізації.

Мета. Розробка алгоритмів високої швидкодії для глобальної оптимізації функції однієї змінної, заснованих на умовах, які дозволяють привести задачу до виду, що відкриває практичну можливість отримання рішення із заданою точністю.

Метод. Розглянуто два алгоритми умовної глобальної оптимізації функції однієї змінної. Перший заснований на оцінці найменшої відстані між сусідніми локальними екстремумами і дозволяє знайти глобальний мінімум цільової функції і при необхідності ці її локальні екстремуми. Другий придатний для пошуку глобального мінімуму функції, якщо наперед відома кількість локальних екстремумів на відрізку невизначеності. Обидва алгоритми базуються на методах сегментації вихідного відрізка невизначеності. Локальний екстремум на сегменті визначається за трьома або по чотирьох точках. Запропоновано підхід, який у більшості випадків дозволяє виконати локалізацію екстремуму по трьох точках, що дає економію при обчисленнях ЦФ, сприяючи тим самим підвищенню швидкодії алгоритму.

Результати. Наведено результати розв'язання оптимізаційних задач і дані про ефективність запропонованих алгоритмів. Проведено порівняльний аналіз швидкодії розроблених алгоритмів і відомих алгоритмів на прикладі рішення тестових завдань, що використовуються у світовій практиці для оцінки ефективності алгоритмів глобальної оптимізації. Наведені приклади практичного використання алгоритмів. Аналіз отриманих даних показав, що за кількістю звернень до цільової функції алгоритми у режимі послідовних обчислень працюють у кілька разів швидше сучасних швидкодіючих алгоритмів, з якими проводилося порівняння.

Висновки. Наведені дані свідчать про ефективність і високу швидкодію запропонованих алгоритмів. Їх швидкодія буде ще вище, якщо викладені ідеї алгоритмізації поширити на паралельні обчислення. Це дозволяє припустити, що запропоновані алгоритми можуть знайти практичне застосування при глобальної оптимізації функцій розглянутих класів задач.

КЛЮЧОВІ СЛОВА: функція однієї змінної, локальний мінімум функції, глобальний мінімум функції, глобальна оптимізація, метод Брента, швидкодія алгоритму.

УДК: 519.67

ДВА АЛГОРИТМА ГЛОБАЛЬНОЙ ОПТИМИЗАЦИИ ФУНКЦИЙ ОДНОЙ ПЕРЕМЕННОЙ, ОСНОВАННЫЕ НА РАССТОЯНИИ МЕЖДУ ЭКСТРЕМУМАМИ И ИХ КОЛИЧЕСТВЕ

Коднянко В. А. – д-р техн. наук, профессор кафедры стандартизации, метрологии и управления качеством. Политехнический институт Сибирского федерального университета, Красноярск, Россия.

АННОТАЦИЯ

Актуальность. Принятие управленческих решений часто связано с решением задач одномерной глобальной оптимизации. Важнейшим свойством методов глобальной оптимизации является их быстродействие, которое определяется количеством обращений к целевой функции в процессе оптимизации.

Цель. Разработка алгоритмов высокого быстродействия для глобальной оптимизации функции одной переменной, основанных на условиях, которые позволяют привести задачу к виду, открывающему практическую возможность получения решения с заданной точностью.

Метод. Рассмотрено два алгоритма условной глобальной оптимизации функции одной переменной. Первый основан на оценке наименьшего расстояния между соседними локальными экстремумами и позволяет найти глобальный минимум целевой функции и при необходимости все ее локальные экстремумы. Второй пригоден для поиска глобального минимума функции, если наперед известно количество локальных экстремумов на отрезке неопределенности. Оба алгоритма базируются на методах сегментации исходного отрезка неопределенности. Локальный экстремум на сегменте определяется по трем или по четырем точкам. Предложен подход, который в большинстве случаев позволяет выполнить локализацию экстремума по трем точкам, что дает экономию при вычислениях ЦФ, способствуя тем самым повышению быстродействия алгоритма.

Результаты. Приведены результаты решения оптимизационных задач и данные об эффективности предложенных алгоритмов. Проведен сравнительный анализ быстродействия разработанных алгоритмов и известных алгоритмов на

DOI 10.15588/1607-3274-2020-2-4

примере решения тестовых задач, используемых в мировой практике для оценки эффективности алгоритмов глобальной оптимизации. Даны примеры практического использования алгоритмов. Анализ полученных данных показал, что по числу обращений к целевой функции алгоритмы в режиме последовательных вычислений работают в несколько раз быстрее современных быстродействующих алгоритмов, с которыми производилось сравнение.

Выводы. Приведенные данные свидетельствуют об эффективности и высоком быстродействии предложенных алгоритмов. Их быстродействие будет еще выше, если изложенные идеи алгоритмизации распространить на параллельные вычисления. Это позволяет предположить, что предложенные алгоритмы могут найти практическое применение при глобальной оптимизации функций рассмотренных классов задач.

КЛЮЧЕВЫЕ СЛОВА: функция одной переменной, локальный минимум функции, глобальный минимум функции, глобальная оптимизация, метод Брента, быстродействие алгоритма.

ЛІТЕРАТУРА / ЛИТЕРАТУРА

1. Bryson A. E., Ho Y. Applied Optimal Control: Optimization, Estimation and Control. / A. E. Bryson, Y. Ho. – Hemisphere Publishing Corporation, Washington DC, 1975. – 477 p.
2. On global extremum seeking in the presence of local extrema / [Y. Tan, D. Nešić, M. Y. Mareels, A. Astolfi] // Automatica. – 2009. – Vol. 45 (1). – P. 245–251. https://doi.org/10.1016/j.automatica.2008.06.010
3. Bizon N. Energy optimization of fuel cell system by using global extremum seeking algorithm. / N. Bizon // Applied Energy. – 2017. – Vol. 206. – P. 458–474. https://doi.org/10.1016/j.apenergy.2017.08.097
4. Bertsekas D. P. Parallel and Distributed Computation: Numerical Methods / D. P. Bertsekas, J. N. Tsitsiklis. – New Jersey, Prentice Hall, 1989. – 94 p.
5. Developments in Global Optimization / [I. M. Bomze, T. Csendes, R. Horst, P. M. Pardalos]. – London, Kluwer, 1997. – 348 p.
6. Квасов Д. Е. Методы липшицевой глобальной оптимизации в задачах управления / Д. Е. Квасов, Я. Д. Сергеев // Автоматика и телемеханика. – 2013. – № 9. – С. 3–19.
7. Сергеев Я. Д. Адаптивные диагональные кривые и их программная реализация / Я. Д. Сергеев, Д. Е. Квасов // Вестник ННГУ: Математическое моделирование и оптимальное управление. – 2001. – Вып. 2, № 24. – С. 300–317.
8. Сергеев Я. Д. Диагональные методы глобальной оптимизации / Я. Д. Сергеев, Д. Е. Квасов. – М. : ФИЗМАТЛИТ, 2008. – 276 с.
9. Pizzuti C. Comparison of two partitionstrategies in diagonal global optimization algorithms: Technical report 9. – Institute of Systems and Informatics / C. Pizzuti, Ya. D. Sergeyev. – Italy, Rende, 2001. – 16 p.
10. Dennis J. E. Jr. Direct Search Methods on Parallel Machines / J. E. Jr. Dennis, V. Torczon // SIAM Journal Optimization. – 1991. – P. 448–474.
11. Nemhauster G. L. Branch-and-bound and Parallel Computation: a Historical Note / G. L. Nemhauster, E. A. Pruul, R. A. Rushmeier // Operations Research Letters. –1988. – No 7. – P. 65–69.
12. Orlyanskaya I. V. An Implementation of the Parallel Continuation Algorithm for Global Optimization / I. V. Orlyanskaya, Y. N. Perunova, S. K. Zavriev // Сборник трудов 3-й Московской международной конференции по исследованию операций. – 2001. – С. 91–92.
13. Ali M. Stochastic Global Optimization: Problem, Classes and Solution Techniques / M. Ali, A. Törn, S. Viitanen // Journal of Global Optimization. – 1999. – No 14. – P. 437–447.
14. Moccus J. Application of Bayesian Approach to Numerical Methods of Global and Stochastic Optimization / J. Moccus // Journal Global Optimization. – 1994. – Vol. 4, No. 4. – P. 347–356.
15. Hansen P. Lipschitz optimization / P. Hansen, B. Jaumard // Handbook of global optimization. – 1995. – Vol 1. – P. 407–493.
16. Коднянко В. А. Оптимизация унимодальных функций методом параболического предиктора / В. А. Коднянко // Вестник Астраханского государственного технического университета. Серия: Управление, вычислительная техника и информатика. – 2018. – № 3. – С. 101–108. DOI: 10.24143/2072-9502-2018-3-101-108
17. Brent R. P. Algorithms for Minimization Without Derivatives / R. P. Brent. – Dover, 2002. – 195 p.
18. Brent R. P. Errata for Algorithms for Minimization without Derivatives / R. P. Brent. – Prentice-Hall edition, 1973. Available from: https://maths-people.anu.edu.au/~brent/pub/pub011_errata.html
19. Burkardt J. V. Brent algorithms for minimization without derivatives, 2008. Available from: http://people.sc.fsu.edu/~jburkardt/f77_src/brent/brent.html
20. Rheinboldt W. C. Algorithm 596: a program for a locally parameterized / W. C. Rheinboldt, J. V.Burkardt. – 2008. Available from: http://people.sc.fsu.edu/~jburkardt/f77_src/brent/brent.f
21. Rao, S. S. Engineering optimization: theory and practice / S. S. Rao. – John Wiley & Sons, Inc., Hoboken, New Jersey, 2009. – 813 p.
22. Amdahl G. The Validity of Single Processor Approach to Achieving Large Scale Computing Capabilities / G. Amdahl // Spring Joint Computer Conference. – 1967. – Vol. 30. – P. 483–485.
23. Nocedal J. Theory of algorithms for unconstrained optimization / Nocedal J. – Acta Numerica, Serles A, Cambridge University Press, Cambridge. – 1991. – P. 199 – 242.
24. Birkhoff S. Piecewise polynomial interpolation and approximation / S. Birkhoff, C. R. de Boor // Garabedian, General Motors Symposium – 1964, Elsevier, New York and Amsterdam. – 1965. – P. 164–190.
25. Epps B. P. Evaluating derivatives of experimental data using smoothing splines / B. P. Epps, T. T. Truscott, A. H. Techet // 3rd Mathematical Methods in Engineering International Symposium MME'10, Coimbra, Portugal. – 2010. – P. 21–24.
26. Kodnyanko V. Quality of Dynamics of Gas-static Thrust Bearing with Movable Carrying Circle on Elastic Suspension / V. Kodnyanko, A. Kurzakov // Tribology in Industry. – 2019. – Vol. 41, No. 2. – P. 237–241. DOI: 10.24874/ti.2019.41.02.09
27. Kurosh A. Higher Algebra / A. Kurosh – M: Mir Publishers, 1984. – P. 432.
28. An algebraic algorithm to isolate complex polynomial zeros using Sturm sequences / M. A. O. Camargo Brunetto, D. M. Claudio, V. Trevisan // Computers & Mathematics with Applications. – 2000. – Vol. 39 (3–4). – P. 95–105. https://doi.org/10.1016/S0898-1221(99)00336-3
29. Vandergraft J. S. Certification of Algorithm 3: Solution of polynomial equations by Bairstow-Hitchcock method / J. S. Vandergraft // Communications of the ACM. – 1961. – Vol. 4, No 2. – P. 105–117.

DOI 10.15588/1607-3274-2020-2-4