절단평면법과 전역 선탐색 알고리즘을 이용한 전역 최적화

박영철, 장민호, 김범수, 이태용
한국과학기술원 생명화학공학과

Global Optimization by Cutting Plane Method and Global Line Search Algorithm

Young Cheol Park, Min Ho Chang, Bum-soo Kim, and Tai-yong Lee
Department of Chemical and Biomolecular Engineering, KAIST

Introduction

Many engineering applications can be formulated as nonlinear function optimization problems in which the function to be optimized possesses many local minima in the parameter region of interest. In most cases, it is desired to find the local minimum at which the function takes its lowest value, i.e., the global optimum. Many a times gradient based schemes gets stuck in a local minimum point which is on the valley of attraction nearest to the starting point. To escape from a local minimum, methods of generalized descent which are classified as heuristic deterministic methods are used. They include the trajectory and penalty methods, such as the tunneling method. The tunneling method was initially developed for unconstrained problems and the basic idea is to execute the following two phases successively until some stopping criterion is satisfied. The minimization phase finds a local minimum point, and the tunneling phase determines another starting point for the local minimization phase with the cost function value smaller than or equal to the known local minimum value. For determining another starting point, DTM(dynamic tunneling method) and TRUST(terminal repeller unconstrained subenergy tunneling) are developed. However, finding a suitable point in the tunneling phase is also a global problem that is as hard as the original problem[2,3,4,6] .

In this paper, we propose new escaping method other than tunneling function. In escaping phase, cutting plane method and global line search algorithm are applied.

Proposed Algorithm

The proposed algorithm can be decomposed into two phases. One is a local optimization phase(LOP) and the other is escape phase(EP). On the LOP, we use the conjugate gradient method. The reason we select this method is that the conjugate gradient method is reliable far from local optimum and accelerates as the sequence of iterates approached the optimum. Additionally, the conjugate direction method is evaluated completely by many authors since it was proposed. Because we use a gradient information, the objective function must be differentiable. We assume that we have a minimization problem.

$$\min \ f(x_i)$$
$$s.t. : \ a_i \le x_i \le b_i, \quad i = 1, \cdots, n$$

Let x_i^0 be a starting point. From the starting point, x_i^0, the conjugate gradient method will find any local optimum, $x_i^{j(*)}$, $f(x_i^{j(*)}) \le f(x_i^j)$. As we know, using local optimization method itself we cannot escape the local optimum. Therefore the EP is needed. On the EP, we solve the new problem which is composed by the following feasibility problem.

$$Find \ x_i \in F^*$$
$$F^* =: \| x_i - x_i^{j(*)} \| \ge \varepsilon$$

$$f(x_i) \le f(x_i^{j(*)})$$
$$a_i \le x_i \le b_i, \quad i=1,\cdots,n$$

The feasible region, F^*, is shown in Fig 1. Once we obtain the local optimum, $x_i^{j(*)}$, we will determine epsilon so that $x_i^{j(*)}$ is the unique optimum in the epsilon-neighborhood of $x_i^{j(*)}$. If we can get x_i^{j+1}, that means we escape the local minimum, because x_i^{j+1} is located outside of epsilon-neighborhood of $x_i^{j(*)}$. Note that x_i^{j+1} is not the local or global optimum but a new starting point. (To calculate a new starting point, in other word, to escape from a local optimum, a novel method based on the cutting plane method and the global line search algorithm is implemented.) At this point of time, we come back to the local optimization phase, then calculate another local optimum, $x_i^{j+1(*)}$, using x_i^{j+1} as a new starting point. These procedures continue until the condition of the termination criterion is satisfied. A simple flowchart which represents a whole procedure is shown in Fig 2.

Since the feasible region , F^*, which is formulated by given constraints is highly nonconvex and nonlinear, it is not easy to get a feasible point. As mentioned before, cutting plane method[7] and global line search algorithm[5] are applied to find a feasible point. For nonlinear objective function, the given problem can be reformulated as

$$Min\ x_0$$
$$s.t: \ g_j(x_i) \ge 0 \quad i=1,\ldots,N \quad j=1,\ldots,J$$
$$x_i^{(L)} \le x_i \le x_i^{(U)} \ i=1,\ldots,N$$
$$x_0 - f(x_i) \ge 0 \quad i=1,\ldots,N$$

The bound for x_0 can be determined by interval analysis. The inequality constraints are imposed to subproblem after linearization using Taylor series.

As we see in flow chart and Fig. 1, this is a very simple and effective method to find the global optimum. In the following section, we evaluated this method by some examples. Thus general cutting plane can be written as

$$p_j^{(i)}(x;x^{(i)}) \equiv g_j(x^{(i)}) + \nabla g_j(x^{(i)})(x - x^{(i)})$$

Let the first solution of LP in cutting plane method be x_i^{LP}. It is natural that this solution violates the constraints of the problem, because this solution is the one when all constraints are neglected. These constraints are

$$g(1) = \|x_i - x_i^{(*)}\| > \varepsilon$$
$$g(2) = f(x_i^{(*)} - f(x_i) \ge 0$$
$$a_i \le x_i \le b_i, \quad i=1,\ldots,N$$

At this time the cutting plane is made based on x_i^{LP}. The most violated constraint is linearized to be constructed as a cutting plane. Then this cutting plane cuts the search region and makes a new search region. During cut, the feasible region may be cut bet this is not a problem. The next step is global line search, a univariate DCEM(difference of convex envelop method), to find a feasible point. In first, to prepare global line search, the line which is perpendicular to the cutting plane and cross the x_i^{LP} is made. The reason we make a perpendicular line to the cutting plane is that it is expected that this perpendicular line crosses much more feasible region than others. Then we calculate two points, a and b, which meet the cutting plane and the boundaries of each variable. The objective function is reformulated as a one dimensional problem as follow.

$$\min_{0\le\alpha\le1} (\vec{A} + \alpha\,(\vec{B} - \vec{A}))$$

Here $\vec{A}$ and $\vec{B}$ are a vector which represents the point a and b. And α is a decision variable. To obtain α, the global line search is used. The detailed procedure is shown in Fig 3 and Fig 4.

The termination criterion in proposed algorithm is the following escape problem does not have any more feasible region.

Examples and Results

We demonstrate the capability of proposed algorithm to find global solutions by solving some multidimensional nonlinear optimization problems. First, describe these problems, which widely used multimodal simple-bounded test functions with known optimal solutions. Then, we report the results of the proposed algorithm and compare the results with those obtained by several generalized descent algorithms.

Example 1) Six-hump camelback function (2-dimension)

$$f(x_1, x_2) = (4 - 2.1x_1^2 + \frac{x_1^4}{3})x_1^2 + x_1x_2 + 4(x_2^2 - 1)x_2$$

$$-2 \le x_1, x_2 \le 2$$

Example 2) Beal function (2-dimension)

$$f(x_1, x_2) = (1.5 - x_1 + x_1x_2)^2 + (2.25 - x_1 + x_1x_2^2)^2 + (2.625 - x_1 + x_1x_2^3)^2$$

$$-1.5 \le x_1 \le 7.5,\ \ -4 \le x_2 \le 5$$

Example 3) Schvefel function(Schw 3.1) (3-dimension)

$$f(x) = \sum_{i=1}^{3}((x_1 - x_i^2)^2 + (x_i - 1)^2)$$

$$-10 \le x_i \le 10,\ \ i = 1, 2, 3$$

Example 4) Powell function (4-dimension)

$$f(x) = (x_1 + 10x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - 2x_3)^4 + 10(x_1 - x_4)^4$$

$$-4 \le x_i \le 5,\ \ i = 1, 2, 3, 4$$

Example 5) Hartman NO. 6 (6-dimension)

The results for the test problems are shown in Table 1. Also, we compared the number of function call with other generalized descent methods for only Ex 1 and the results are shown in Table 2. In the table, TM is tunneling method, DTM is dynamic tunneling method, and TRUST is Terminal repeller unconstrained subenergy tunneling[6].

Conclusion

The new algorithm with the suggested escape phase is successfully applied to the general nonlinear problems of unconstrained global optimization. The proposed algorithm shows good performance to several standard test problems. In a view point of number of function calls, proposed algorithm reduces them by a factor of 10 to 100 when it is compared with interval branch and bound method. It is expected that when compared with other stochastic methods, the degree of reduction will increase. Also, the suggested algorithm performs better when it is applied to large size problems which have many independent variables.

Acknowledgement

This work was also partially supported by the Brain Korea 21 project.

References

1. Bum-soo Kim, "Global Optimization by Successive Local optimizaton and Escape", Master thesis, KAIST, 2002
2. O. A. Elwakeil and J. S. Arora, "Two Algorithm for Global Optimization of General NLP Problems", *International Journal for Numerical Methods in Eng., vol. 39*, 1996.
3. J. S. Arora, O. A. Elwakeil, A. I. Chahande and C. C. Hsieh, "Global Optimization Methods for Engineering Applications: a review", *Struct. Optim. vol. 9*, pp. 137-159, 1995
4. Pinaki RoyChowdhury, Y. P. Singh, and R. A. Chansarkar, "Hybridization of Gradient Descent Algorithms with Dynamic Tunneling Methods for Golbal Optimization", *IEEE Transactions on Systems, Man, and Cybernetics - Part A, vol 30, No. 3*, May 2000
5. Young Cheol Park, Min Ho Chang and Tai-yong Lee, "Constrained Univariate Global Optimization Algorithm using D.C. Underestimator", *Theories and Applications of Chem. Eng., 2002, Vol. 7, No. 2*, 2001
6. B. C. Cetin, J. Barhen, and J. W. Burdick, "Terminal Repeller Unconstrained Subenergy Tunneling(TRUST) for Fast Global Optimization", *J. Optimization Theory and Applications, vol. 77, No. 1*, April 1993
7. G. V. Reklaitis, A. Ravindran, K. M. Ragsdell, "Engineering Optimization Methods and Applications", John Wiley and Sons, pp 327-340, 1983

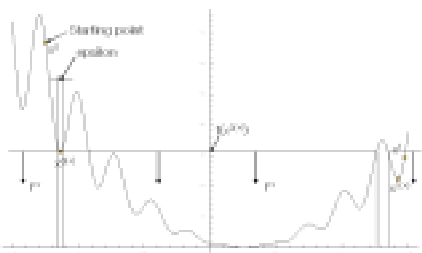

Fig 1. The overall scheme

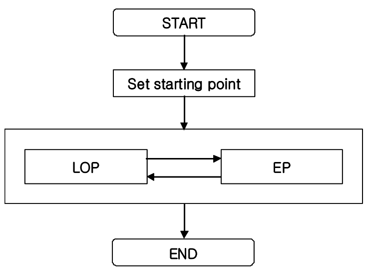

Fig 2. Overall flowchart

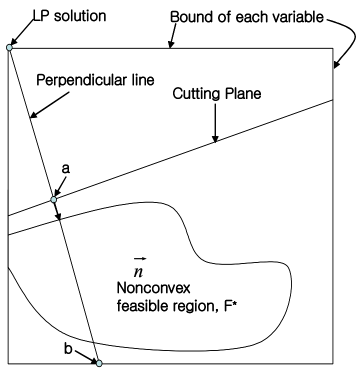

Fig 3. Line search in EP

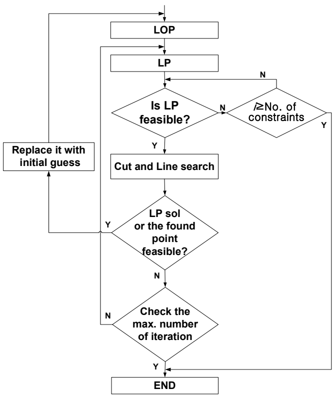

Fig 4. EP flowchart

	Bound of x_i	Starting point	Best known solution	Obtained solution
Ex 1.	[-3, 3]	$x_i = 0$	(−0.0897, 0.7127)	(−0.0899, 0.7127)
Ex 2.	x_1=[-1.5, 7.5] x_2=[-4, 5]	$x_1 = 3$ $x_2 = 1$	(3, 0.5)	(2.9998, 0.4999)
Ex 3.	[-10, 10]	$x_i = 0$	(1, 1, 1)	(1.0009, 1.0002, 1.0002)
Ex 4.	[-4, 5]	$x_i = 3$	(0, 0, 0, 0)	(−3.8×10[illegible], 3.84×10[illegible], −5.25×10[illegible], −5.28×10[illegible])
Ex 5.	[0, 10]	$x_i = 5$	(0.201690, 0.150011, 0.476874, 0.275332, 0.311652, 0.657300)	(0.201615, 0.150026, 0.476884, 0.257305, 0.311433, 0.657293)

Table 1. Results for test problems

Starting point	Methods			
	TM	DTM	TRUST	Proposed algorithm
(-3, 2)	1496	1496	168	151
(3, 2)	1496	1132	168	127
(-2, -1)	1496	n/a	32	146
(-1.6, 0.9)	n/a	n/a	76	122

Table 2. # of function calls