MinCenter: using clustering in global optimization

Abstract

A common problem arises in many scientific fields is that of locating the global minimum of a multimodal function. A novel clustering technique that tackles this problem is introduced here. The proposed method creates clusters from uniform samples of the objective function with the usage of the Kmeans clustering technique. For every cluster a center is created. Finally, a simple rejection procedure is applied to the created clusters in order to remove clusters that are close to others. The proposed method is tested on a series of well - known optimization problems from the relevant literature and the results are reported and compared against the simple Multistart global optimization method.

Keywords: Global optimization, clustering, hubrid methods, numerical methods.

1 Introduction

A novel method that estimates the global minimum of a continuous and differentiable function $f : S \to R, S \subset R^n$ is proposed in the current article. The global optimum location problem is usually defined as:

$$x^* = \arg\min_{x \in S} f(x) \tag{1}$$

where S is

$$S = [a_1, b_1] \otimes [a_2, b_2] \otimes \ldots [a_n, b_n]$$

A review of the recent advantages in the area of the Global Optimization can be found in [1]. Methods that discover the global minimum can be used in many areas such as: economics [2, 3], physics [4, 5], chemistry [6, 7], medicine [8, 9] etc. Global optimization methods usually are divided into two main categories: deterministic and random search methods. Common methods of the first category are the so called Interval methods [10, 11], where the set S is divided iteratively in subregions using some criteria. On the other hand, random search methods are used in the majority of cases, because they can be implement easy and they do not depend on a some a priori information about the objective function. A small set of random search methods may include Controlled Random Search methods [12, 13, 14], Simulated Annealing methods [15, 16], Differential Evolution methods [17, 18], Particle Swarm Optimization methods [19, 20], Ant Colony Optimization [21, 22], Genetic algorithms [23, 24, 25] etc.

A subclass of random search methods are the clustering techniques as proposed by Rinnooy Kan [26], Ali [27], Tsoulos [28], etc. These methods are try to estimate the clusters of function in order to minimize the effort required to compute the global minimum or all the local minima of the function. The term cluster refers to a set of points that are believed, under some asymptotic considerations, to belong to the same region of attraction of the function. The region of attraction for a local minimum x^* is defined as:

$$A\left(x^*\right) = \left\{x:\ x \in S \subset R^n,\ L(x) = x^*\right\} \tag{2}$$

where $L(x)$ is a local search procedure that starts from a given point x and terminates when a local minimum is discovered. Common local search procedures are BFGS[29, 30], Steepest Descent[31], L-Bfgs [32] for large scaled functions etc. The proposed method creates clusters iteratively using the well - known technique of the K-Means clustering introduced by MacQueen[33]. For every cluster a representative is constructed using the K-means method and afterwards a rejection procedure is utilized in order to reduce the number of representatives. Finally, for every remain point a local search procedure is started to locate the global minimum of the function.

The rest of this article is organized as follows: in section 2 the proposed method is described in detail, in section 3 some experimental test functions from the relevant literature are described and a series of test are performed on those functions and finally in section 4 some conclusions are discussed as well as some guidelines to improve the proposed method.

2 Method description

The proposed method is initially based on the commonly used global optimization method named Multistart. The proposed method creates clusters from the objective function. The multistart method is one of the simplest global optimization technique which start a local search optimizer such as BFGS from different random points and yields the lowest discovered minimum as the global one. As it was demonstrated by various researchers [34, 35], if the number of local minimum is finite then Multistart method is capable to locate the global minimum. Due to its simplicity, the Multistart method is the base method for a series of stochastic methods in the relevant literature such as hybrid methods[36, 37], GRASP methods[38] etc. The main steps of a typical Multistart procedure are shown in Algorithm 1.

The proposed method replaces the sampling step of the Multistart method with the usage of centroids constructed by Kmeans clustering. The main steps of the Kmeans method are given in Algorithm 2. The estimated centroids

Algorithm 1 The main steps of the Multistart method.

1. **Initialization** step.
   (a) **Set** M as the total number of samples.
   (b) **Set** (x^*, y^*) as the global minimum. Initialize y^* to a very large value.
2. **Sampling** step.
   (a) **For** $i = 1 \ldots M$ **Do**
      i. **Sample** a point $x_i \in S$
      ii. $y_i = \mathrm{LS}(x_i)$. Where LS(x) is a local search procedure.
      iii. **If** $y_i \leq y^*$ then $x^* = x_i, y^* = y_i$
   (b) **EndFor**

are iteratively enhanced with Kmeans and new samples that added each time for a predefined number of times. Having created the centroids a rejection procedure is applied to reduce the number of centroids. The rejection procedure removes from the set of centers, points that have many neighbors in a predefined radius. The rejection procedure is necessary to remove from the set samples, that possible will yeld the same local optimum after the application of the local search procedure. The proposed method is described in Algorithm 3.

3 Experiments

3.1 Test functions

In order to measure the effectiveness of the proposed approach we utilize several benchmark functions from the relevant literature [39, 40]. The names and the complete forms of these equations can be obtained from [41].

3.2 Experimental Results

The proposed method was tested against the traditional multistart global optimization method on the series of benchmark problems. The parameters used in the conducted experiments are listed in Table 1. The method was coded using the OpenMP library[42] in order to take advantage of multi - core modern computing systems and the experiments were conducted on a cluster of systems running the Linux operating system. The results from the experiments are listed in Tables 2 and 3. The methods were executed 30 times for each test function and averages were measured. In every run different seed for the random generator (drand48() function of C) was used. The numbers appeared in the parentheses stand for the percentage of runs where the global minimum was

Algorithm 2 The algorithm Kmeans.

1. **Repeat**
   (a) $S_j = \{\}\,,\ j = 1..K$
   (b) **For** every sample x_i **Do**
      i. **Set** $j^* = \min_{i=1}^{K}\{D(x_i, c_j)\}$, where j^* is the nearest center for sample x_i.
      ii. **Set** $S_{j^*} = S_{j^*} \cup \{x_i\}$.
   (c) **EndFor**
   (d) **For** every center c_j **Do**
      i. **Set** M_j=number of elements in S_j
      ii. **Update** c_j

$$c_j = \frac{1}{M_j}\sum_{i=1}^{M_j} x_i$$

   (e) **EndFor**
2. **Terminate** when c_j no longer change.

located sucessfully and they are not shown if the success was 100%. The last row represents the total number of function calls. The local search procedure used (denotes as LS(x)) was a BFGS variant due to Powell[30].

In Table 2 the results for the Multistart global optimization procedure are shown. The column $M = 100$ denotes the application of the algorithm given in Algorithm 1 with $M = 100$ samples. The column $M = 200$ stands for the Multistart algorithm using 200 samples. The last column stands for the results of the Multistart method with 100 samples and the application of the proposed rejection procedure of algorithm in Algorithm 4 in the samples before the application of the local search procedure. It is evident that the application of the rejection procedure does not reduce significantly the number of function calls for the multistart case.

In Table 3 the experimental results for the proposed method are listed. The column $M = 100$ stands for the usage of 100 samples in the proposed method (parameter M) and the column $M = 200$ for 200 samples. The proposed method has significantly lower number of function calls than the Multistart method and as the number of samples increases (parameter M) the method requires lower amount of function calls to estimate the global minimum. This means that the method tends to create more accurate clusters (clusters that emulate the true regions of attraction) of the objective function as the number of samples increases.

Algorithm 3 The proposed method.

1. **Initialization** step.
   (a) **Set** M as the number of samples.
   (b) **Set** (x^*, y^*) as the global minimum. Initialize y^* to a very large value.
   (c) **Set** K the number of teams, where $K < M$.
   (d) **Set** K_{MAX} the number of construction iterations for the KMeans algorithm.
   (e) **Set** $C = \{\}$, as the set of constructed centers.
2. **Construction** step.
   (a) **For** $i = 1..K_{\text{MAX}}$ **Do**
      i. **Sample** M points from the objective function $S = \{x_1, x_2, \ldots, x_M\}$
      ii. **Update** the centers C with the set S, using Kmeans.
   (b) **EndFor**
3. **Create** the set R from C using the rejection algorithm of Algorithm 4.
4. **Evaluation** step.
   (a) **For** $i = 1 \ldots |R|$ **Do**
      i. **Set** $x_i = R_i$
      ii. $y_i = \text{LS}(x_i)$. Where LS(x) is a local search procedure.
      iii. **If** $y_i \leq y^*$ then $x^* = x_i, y^* = y_i$
   (b) **EndFor**

Table 1: The values for the parameters used in the conducted experiments.

PARAMETER	VALUE
K	100
K_{max}	100
F	1.5
N_{min}	3

Table 2: Multistart results.

Function	$M = 100$	$M = 100$	$M = 100$,Rejection
B2	4518	8849	4472
Easom	943	1949	933
Bf1	4508	9300	4469
Bf2	3750	7621	3666
Branin	1948	3855	1938
Camel	2669	4983	2502
CM4	5714	11783	5644
CM8	7341(0.33)	14813(0.60)	7289(0.33)
DIFFPOWER10	123729	248924	121012
ELP4	1203	2474	1158
ELP8	1721	3395	1652
ELP16	2789	5485	2252
EXP4	3646	7063	3609
EXP8	3723	7447	3651
EXP16	3835	7486	3310
GKLS250	1486	2928	1426
GKLS350	1030(0.97)	2007	913(0.87)
GKLS3100	1020(0.77)	2005	1018(0.77)
GRIEWANK2	3131(0.70)	6197(0.97)	3048(0.70)
GRIEWANK10	10449	20763	10226
HANSEN	2482	4997	2422
HARTMAN3	2911	5753	2868
HARTMAN6	3825	7875	3787
POTENTIAL3	5237	10784	5178
POTENTIAL5	11594	22331	10127
POTENTIAL10	20361	40592	5089(0.70)
RASTRIGIN	2345	4731	2242(0.93)
SHEKEL5	3852	7841	3730
SHEKEL7	3951	7149	3885
SHEKEL10	3982	6987	3890
SINU4	3317	6624	3246
SINU8	4883	10015	4791
SINU16	8731	17005	8692
TEST2n4	3258	6608	3216
TEST2n5	3565	7128	3534
TEST2n6	3804(0.90)	7790	3850(0.90)
TEST2n7	4203(0.83)	8501(0.97)	4155(0.77)
TOTAL	**281454(0.93)**	**562038(0.98)**	**259160(0.92)**

Table 3: The proposed method with $K = 100$ centers.

Function	$M = 100$	$M = 200$
B2	4073	3886
Easom	830	782
Bf1	4046	3864
Bf2	3346	3153
Branin	1699	1623
Camel	2338	2237
CM4	4434	4043
CM8	3084(0.63)	1819(0.50)
DIFFPOWER10	26726	17980
ELP4	971	908
ELP8	601	338
ELP16	139	100
EXP4	2764	2522
EXP8	1564	943
EXP16	245	179
GKLS250	1337	1275
GKLS350	911(0.93)	777(0.83)
GKLS3100	939(0.97)	796(0.97)
GRIEWANK2	2812(0.77)	2684(0.70)
GRIEWANK10	1812	1152(0.80)
HANSEN	2210	2077
HARTMAN3	2400	1993
HARTMAN6	2707	2369
POTENTIAL3	1246	714
POTENTIAL5	752	664
POTENTIAL10	1621(0.23)	1045(0.10)
RASTRIGIN	2016	1917
SHEKEL5	3520	3116
SHEKEL7	3515	3113
SHEKEL10	3586	3237
SINU4	2548	2268
SINU8	2121	1624
SINU16	546	358
TEST2n4	2436	2198
TEST2n5	2186(0.97)	1840(0.97)
TEST2n6	2648(0.80)	2300(0.83)
TEST2n7	2469(0.77)	2173(0.73)
TOTAL	**103198(0.95)**	**84067(0.93)**

Algorithm 4 The rejection algorithm.

1. **Set** C the set of centers.
2. **Set** $R = \emptyset$ the outcome of the rejection algorithm.
3. **Set** $D_{\text{min}} = \min_{i \neq j} \|c_i - c_j\|$
4. **Set** $F > 1$, a double value.
5. **Set** $N_{\text{min}} > 1$, an integer value.
6. **For** every center c_i **Do**
   (a) **Set** $N = 0$
   (b) **For** every center $c_j,\ i \neq j$ **Do**
      i. **If** $\|c_i - c_j\| \leq FD_{\text{min}}$ **then** $N = N + 1$
   (c) **EndFor**
   (d) **If** $N < N_{\text{min}}$ **then** $R = R \cup c_i$
7. **EndFor**
8. **Return** R

4 Conclusions

A new clustering method was introduced in this article to tackle to global optimization problem. For every cluster gradually a representative is created using the well - known Kmeans method. Afterwards, the clusters are reduced in number using a simple rejection procedure. The proposed method was tested on a series of benchmark problems from the relevant literature and it is compared against the Multistart method and the results are reported. Judging from the reported results, the proposed method seems to be very promising and a series of enhancements could be applied on the method such as:

1. Dynamic selection of K in Kmeans algorithm.
2. Better estimation of the critical distance between clusters in the rejection procedure.
3. Usage of more efficient stopping rules to prevent the method from unnecessary local searches, that could lead to the same global optimum many times.

Compliance with Ethical Standards

All authors declare that they have no has no conflict of interest.

References

[1] C. A. Floudas and C. E. Gounaris, A review of recent advances in global optimization, Journal of Global Optimization 45, pp. 3-38, 2009.

[2] Zwe-Lee Gaing, Particle swarm optimization to solving the economic dispatch considering the generator constraints, IEEE Transactions on **18** Power Systems, pp. 1187-1195, 2003.

[3] C. D. Maranas, I. P. Androulakis, C. A. Floudas, A. J. Berger, J. M. Mulvey, Solving long-term financial planning problems via global optimization, Journal of Economic Dynamics and Control **21**, pp. 1405-1425, 1997.

[4] Q. Duan, S. Sorooshian, V. Gupta, Effective and efficient global optimization for conceptual rainfall-runoff models, Water Resources Research **28**, pp. 1015-1031 , 1992.

[5] P. Charbonneau, Genetic Algorithms in Astronomy and Astrophysics, Astrophysical Journal Supplement **101**, p. 309, 1995

[6] A. Liwo, J. Lee, D.R. Ripoll, J. Pillardy, H. A. Scheraga, Protein structure prediction by global optimization of a potential energy function, Biophysics **96**, pp. 5482-5485, 1999.

[7] P.M. Pardalos, D. Shalloway, G. Xue, Optimization methods for computing global minima of nonconvex potential energy functions, Journal of Global Optimization **4**, pp. 117-133, 1994.

[8] Eva K. Lee, Large-Scale Optimization-Based Classification Models in Medicine and Biology, Annals of Biomedical Engineering **35**, pp 1095-1109, 2007.

[9] Y. Cherruault, Global optimization in biology and medicine, Mathematical and Computer Modelling **20**, pp. 119-132, 1994.

[10] M.A. Wolfe, Interval methods for global optimization, Applied Mathematics and Computation **75**, pp. 179-206, 1996.

[11] T. Csendes and D. Ratz, Subdivision Direction Selection in Interval Methods for Global Optimization, SIAM J. Numer. Anal. **34**, pp. 922–938, 1997.

[12] W. L. Price, Global optimization by controlled random search, Journal of Optimization Theory and Applications **40**, pp. 333-348, 1983.

[13] Ivan Křivý, Josef Tvrdík, The controlled random search algorithm in optimizing regression models, Computational Statistics & Data Analysis **20**, pp. 229-234, 1995.

[14] M.M. Ali, A. Törn, and S. Viitanen, A Numerical Comparison of Some Modified Controlled Random Search Algorithms, Journal of Global Optimization **11**,pp. 377–385,1997.

[15] L. Ingber, Very fast simulated re-annealing, Mathematical and Computer Modelling **12**, pp. 967-973, 1989.

[16] R.W. Eglese, Simulated annealing: A tool for operational research, Simulated annealing: A tool for operational research **46**, pp. 271-281, 1990.

[17] R. Storn, K. Price, Differential Evolution - A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces, Journal of Global Optimization **11**, pp. 341-359, 1997.

[18] J. Liu, J. Lampinen, A Fuzzy Adaptive Differential Evolution Algorithm. Soft Comput **9**, pp.448–462, 2005.

[19] Riccardo Poli, James Kennedy kennedy, Tim Blackwell, Particle swarm optimization An Overview, Swarm Intelligence **1**, pp 33-57, 2007.

[20] Ioan Cristian Trelea, The particle swarm optimization algorithm: convergence analysis and parameter selection, Information Processing Letters **85**, pp. 317-325, 2003.

[21] M. Dorigo, M. Birattari and T. Stutzle, Ant colony optimization, IEEE Computational Intelligence Magazine **1**, pp. 28-39, 2006.

[22] K. Socha, M. Dorigo, Ant colony optimization for continuous domains, European Journal of Operational Research 185, pp. 1155-1173, 2008.

[23] D. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley Publishing Company, Reading, Massachussets, 1989.

[24] Z. Michaelewicz, Genetic Algorithms + Data Structures = Evolution Programs. Springer - Verlag, Berlin, 1996.

[25] S.A. Grady, M.Y. Hussaini, M.M. Abdullah, Placement of wind turbines using genetic algorithms, Renewable Energy **30**, pp. 259-270, 2005.

[26] A.H.G. Rinnooy Kan and G.T. Timmer, Stochastic global optimization methods, Part II: Multilevel methods, Math. Programm. **39** , pp. 57–78, 1987.

[27] M.M. Ali and C. Storey, Topographical multilevel single linkage, Journal of Global Optimization **5**, pp. 349–358, 1994.

[28] I. G. Tsoulos and I. E. Lagaris, MinFinder: Locating all the local minima of a function, Computer Physics Communications **174**, pp. 166-179, 2006.

[29] R. Fletcher, A new approach to variable metric algorithms, Comput. J. **13**, pp. 317–322, 1970.

[30] M.J.D Powell, A Tolerant Algorithm for Linearly Constrained Optimization Calculations, Mathematical Programming **45**, pp. 547-566, 1989.

[31] Ya-xiang Yuan, A new stepsize for the steepest descent method, Journal of Computational Mathematics 24, pp. 149-156, 2006.

[32] Ciyou Zhu, Richard H. Byrd, Peihuang Lu, and Jorge Nocedal, Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization. ACM Trans. Math. Softw. **23**, pp. 550–560, 1997.

[33] MacQueen, J.: Some methods for classification and analysis of multivariate observations, in: Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, Vol. 1, No. 14, pp. 281-297, 1967.

[34] C.G.E. Boender, A.H.G. Kan Rinnooy, Bayesian stopping rules for multistart global optimization methods, Math. Program. **37,** pp. 59–80, 1987.

[35] I.E. Lagaris, I.G. Tsoulos, Stopping rules for box-constrained stochastic global optimization, Applied Mathematics and Computation **197**, pp. 622–632, 2008.

[36] M. Perez, F. Almeida and J. M. Moreno-Vega, "Genetic algorithm with multistart search for the p-Hub median problem," Proceedings. 24th EUROMICRO Conference (Cat. No.98EX204), Vasteras, Sweden, 1998, pp. 702-707 vol.2.

[37] H. C. B. d. Oliveira, G. C. Vasconcelos and G. B. Alvarenga, "A Multi-Start Simulated Annealing Algorithm for the Vehicle Routing Problem with Time Windows," 2006 Ninth Brazilian Symposium on Neural Networks (SBRN'06), Ribeirao Preto, Brazil, 2006, pp. 137-142.

[38] Festa P., Resende M.G.C. (2009) Hybrid GRASP Heuristics. In: Abraham A., Hassanien AE., Siarry P., Engelbrecht A. (eds) Foundations of Computational Intelligence Volume 3. Studies in Computational Intelligence, vol 203. Springer, Berlin, Heidelberg.

[39] M. Montaz Ali, Charoenchai Khompatraporn, Zelda B. Zabinsky, A Numerical Evaluation of Several Stochastic Algorithms on Selected Continuous Global Optimization Test Problems, Journal of Global Optimization **31**, pp 635-672, 2005.

[40] C.A. Floudas, P.M. Pardalos, C. Adjiman, W. Esposoto, Z. Gümüs, S. Harding, J. Klepeis, C. Meyer, C. Schweiger, Handbook of Test Problems in Local and Global Optimization, Kluwer Academic Publishers, Dordrecht, 1999.

[41] Tsoulos, I.G., Karvounis, E. & Tzallas, A. A Novel Sampling Technique for Multistart-Based Methods. SN COMPUT. SCI. 2, 7 (2021). https://doi.org/10.1007/s42979-020-00392-9

[42] R. Chandra, L. Dagum, D. Kohr, D. Maydan,J. McDonald and R. Menon, Parallel Programming in OpenMP, Morgan Kaufmann Publishers Inc., 2001.