An improved Controlled Random Search method

Vasileios Charilogis(1), Ioannis G. Tsoulos(1), Alexandros Tzallas,(1)
Nikolaos Anastasopoulos(2)

(1)Department of Informatics and Telecommunications, University of Ioannina, 47100 Arta, Greece
(2)Department of Electrical and Computer Engineering, University of Patras, Greece

Abstract

Write some abstract here

1 Introduction

Global optimization[1] is considered a problem of high complexity with many applications. The problem is defined as the location of the global minimum of a multi - dimensional function $f(x)$:

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

Where $S \subset R^n$ is formulated as:

$$S = [a_1, b_1] \otimes [a_2, b_2] \otimes \ldots [a_n, b_n] \tag{2}$$

This problem is applicable in a wide range of areas such as physics [2, 3], chemistry [6, 7], medicine [4, 5], economics[8] etc. In modern theory there are two different categories of global optimization methods, the stochastic methods and the deterministic methods. The first category contains the vast majority of methods such as Simulated Annealing methods [9, 10, 11], Genetic Algorithms [12, 13, 14], Tabu Search methods [15], Particle Swarm Optimization [16, 17, 18] etc. A common method that also belongs to stochastic methods is the Controlled Random Search (CRS) method [19], which is a population based algorithm. This method initially creates a set with randomly selected points and repeatedly replaces the worst point in that set with a randomly generated point and this process can continue until some convergence criterion is met. The CRS method has been used intensively in many problems such as geophysics problems [20, 21], optimal shape design problems [22], the animal diet problem [23], the heat transfer problem [24] etc.

This CRS method has been thoroughly analyzed by many researchers in the field such as the work A of Ali and Storey where two new variants of the CRS method are proposed [25]. These variants have proposed alternative techniques for the selection of the initial sample set and usage of local search methods. Also, Pillo *et al* [26] suggested a hybrid CRS method where the base algorithm is combined with a Newton - type unconstrained minimization algorithm [27] to encance the efficiency of the method in various test problems. Another work is of Kaelo and Ali , in which they suggested [28] some modifications on the method and especially in the new point generation step. Also, Filho and Albuquerque have suggested [29] the usage of a distribution strategy to accelerate the Controlled Random Search method. The current work proposed three major modifications in the CRS method: a new point replacement strategy, a stochastic termination rule and the a periodical application of some local search method. The first modification is used to better explore the domain range of the function. The second modification is made in order to achieve a better termination of the method without wasting valuable computational time. The third modification is used in order to speed up the method by applying a small amount of steps of a local search method.

The rest of this article is organized as follows: in Section 2 the major steps of the CRS method as well as the proposed modifications are outlined, in section 3 the results from the application of the proposed method on a series of test functions are provided and finally in section 4 some conclusions are presented.

2 Method Description

The Controlled Random Search a series of steps that are described bellow.

Initialization Step:

1. **Se**t the value for the parameter N. A commonly used value for that is $N = 25n$.

2. **Set** a small positive value for ϵ.

3. **Create** the set $T = \{z_1, z_2, ..., z_N\}$, by randomly sampling N points from S.

Min_Max Step:

1. **Calculate** the points $z_{\text{min}} = \text{argmin} f(z)$ and $z_{\text{max}} = \text{argmax} f(z)$ and their function values

$$f_{\text{max}} = \max_{z \in T} f(z)$$

and

$$f_{\text{min}} = \min_{z \in T} f(z)$$

2. **If** $|f_{\text{max}} - f_{\text{min}}| < \epsilon$, **then goto Local_Search** Step.

New_Point Step:

1. **Select** randomly the reduced set $\tilde{T} = \left\{z_{T_1}, z_{T_2}, .., z_{T_{n+1}}\right\}$ from T.
2. **Compute** the centroid G:

$$G = \frac{1}{n}\sum_{i=1}^{n} z_{T_i}$$

3. **Compute** a trial point $\tilde{z} = 2G - z_{T_{n+1}}$.
4. **If** $\tilde{z} \notin S$ **or** $f(\tilde{z}) \geq f_{\max}$ **then** goto New_Point step.

Update Step:

1. $T = T \cup \{\tilde{z}\} - \{z_{\max}\}$.
2. **Goto Min_Max** Step.

Local_Search Step:

1. $z^* = \mathrm{localSearch}(z)$.
2. Return the point z^* as the discovered global minimum.

2.1 A new method for trial points

1. **Compute** the centroid G:

$$G = \frac{1}{n}\sum_{i=1}^{n} z_{T_i}$$

2. Set $G = G + \frac{1}{n}x_{\min}$
3. **Compute** a trial point $\tilde{z} = G - \frac{1}{n}z_{T_{n+1}}$.

2.2 A new stopping rule

A common way to terminate a global optimization procedure is to use the maximum number of allowed iterations, i.e. stop when iter $\geq K$. Although, it is a simple criterion but is not an efficient one since, if K is tool small, then the algorithm will terminate without locating the global optimum. Also, when K is too high, the optimization algorithm will spend computation time in unnecessary function calls. The termination rule used here is derived from [30]: denote with $\sigma^{(k)}$ the variance of $f\left(x^*\right)$, where k is iteration number and x^* is the so far located global minimum. If the algorithm did not manage to locate new minimum for a number of generations, then probably the

algorithm has located the global minimum and it should terminate. The termination rule stops the algorithm when

$$k \geq k_{\textbf{min}} \ \textbf{AND} \ \sigma^{(k)} \leq \frac{\sigma^{\left(k_{\textbf{last}}\right)}}{2} \tag{3}$$

where $k_{\textbf{last}}$ stands for the iteration where a new minimum was lastly found. The value $k_{\textbf{min}}$ is a predefined minimum number of iterations used to prevent the algorithm from premature termination.

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 [31, 32]. The following functions were used:

- **Bf1** function. The function Bohachevsky 1 is given by the equation

$$f(x) = x_1^2 + 2x_2^2 - \frac{3}{10}\cos\left(3\pi x_1\right) - \frac{4}{10}\cos\left(4\pi x_2\right) + \frac{7}{10}$$

with $x \in [-100, 100]^2$. The value of global minimum is 0.0.

- **Bf2** function. The function Bohachevsky 2 is given by the equation

$$f(x) = x_1^2 + 2x_2^2 - \frac{3}{10}\cos\left(3\pi x_1\right)\cos\left(4\pi x_2\right) + \frac{3}{10}$$

  with $x \in [-50, 50]^2$. The value of the global minimum is 0.0.

- **Branin** function. The function is defined by $f(x) = \left(x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos(x_1) + 10$ with $-5 \leq x_1 \leq 10, \ 0 \leq x_2 \leq 15$. The value of global minimum is 0.397887.with $x \in [-10, 10]^2$. The value of global minimum is -0.352386.

- **CM** function. The Cosine Mixture function is given by the equation

$$f(x) = \sum_{i=1}^{n} x_i^2 - \frac{1}{10}\sum_{i=1}^{n}\cos\left(5\pi x_i\right)$$

  with $x \in [-1, 1]^n$. The value of the global minimum is -0.4 and in our experiments we have used $n = 4$.

- **Camel** function. The function is given by

$$f(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4, \quad x \in [-5, 5]^2$$

  The global minimum has the value of $f\left(x^*\right) = -1.0316$

- **Easom** function. The function is given by the equation

$$f(x) = -\cos(x_1)\cos(x_2)\exp\left((x_2-\pi)^2-(x_1-\pi)^2\right)$$

with $x \in [-100, 100]^2$. The value of the global minimum is -1.0

- **Exponential** function. The function is given by

$$f(x) = -\exp\left(-0.5\sum_{i=1}^{n} x_i^2\right), \quad -1 \le x_i \le 1$$

The global minimum is located at $x^* = (0, 0, ..., 0)$ with value -1. In our experiments we used this function with $n = 2, 4, 8, 16, 32, 64, 100$ and the corresponding functions are denoted by the labels EXP2, EXP4, EXP8, EXP16, EXP32, EXP64, EXP100.

- **GoldStein & Price**

$$\begin{aligned} f(x) \quad = \quad & [1 + (x_1 + x_2 + 1)^2 \\ & (19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)] \times \\ & [30 + (2x_1 - 3x_2)^2 \\ & (18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)] \end{aligned}$$

The function has 4 local minima in the range $[-2, 2]^2$ and global minimum $f^* = 3.0$.

- **Griewank2** function. The function is given by

$$f(x) = 1 + \frac{1}{200}\sum_{i=1}^{2} x_i^2 - \prod_{i=1}^{2}\frac{\cos(x_i)}{\sqrt{(i)}}, \quad x \in [-100, 100]^2$$

The global minimum is located at the $x^* = (0, 0, ..., 0)$ with value 0.

- **Gkls** function. $f(x) = \text{Gkls}(x, n, w)$, is a function with w local minima, described in [33] with $x \in [-1, 1]^n$ and n a positive integer between 2 and 100. The value of the global minimum is -1 and in our experiments we have used $n = 2, 3$ and $w = 50$. The corresponding functions are denoted by the labels GKLS250 and GKLS350.

- **Guilin Hills** function. $f(x) = 3 + \sum_{i=1}^{n}\left(c_i \frac{x_i+9}{x_i+10}\sin\left(\frac{\pi}{1-x_i+\frac{1}{2k_i}}\right)\right)$, $x \in [0, 1]^n$, $c_i > 0$ and k_i being positive integers. The function has $\prod_{i=1}^{n} k_i$ local minima. In our experiments we have used $n = 5, 10$ and we set the values of k_i so that the number of minima was 50. The produced functions are entititled GUILIN550 and GUILIN1050 in the result tables.

- **Hansen** function. $f(x) = \sum_{i=1}^{5} i\cos[(i-1)x_1 + i]\sum_{j=1}^{5} j\cos[(j+1)x_2 + j]$, $x \in [-10, 10]^2$. The global minimum of the function is -176.541793.

- **Hartman 3** function. The function is given by

$$f(x) = -\sum_{i=1}^{4} c_i \exp\left(-\sum_{j=1}^{3} a_{ij}\left(x_j - p_{ij}\right)^2\right)$$

with $x \in [0,1]^3$ and $a = \begin{pmatrix} 3 & 10 & 30 \\ 0.1 & 10 & 35 \\ 3 & 10 & 30 \\ 0.1 & 10 & 35 \end{pmatrix}$, $c = \begin{pmatrix} 1 \\ 1.2 \\ 3 \\ 3.2 \end{pmatrix}$ and

$$p = \begin{pmatrix} 0.3689 & 0.117 & 0.2673 \\ 0.4699 & 0.4387 & 0.747 \\ 0.1091 & 0.8732 & 0.5547 \\ 0.03815 & 0.5743 & 0.8828 \end{pmatrix}$$

The value of global minimum is -3.862782.

- **Hartman 6** function.

$$f(x) = -\sum_{i=1}^{4} c_i \exp\left(-\sum_{j=1}^{6} a_{ij}\left(x_j - p_{ij}\right)^2\right)$$

with $x \in [0,1]^6$ and $a = \begin{pmatrix} 10 & 3 & 17 & 3.5 & 1.7 & 8 \\ 0.05 & 10 & 17 & 0.1 & 8 & 14 \\ 3 & 3.5 & 1.7 & 10 & 17 & 8 \\ 17 & 8 & 0.05 & 10 & 0.1 & 14 \end{pmatrix}$, $c = \begin{pmatrix} 1 \\ 1.2 \\ 3 \\ 3.2 \end{pmatrix}$

and

$$p = \begin{pmatrix} 0.1312 & 0.1696 & 0.5569 & 0.0124 & 0.8283 & 0.5886 \\ 0.2329 & 0.4135 & 0.8307 & 0.3736 & 0.1004 & 0.9991 \\ 0.2348 & 0.1451 & 0.3522 & 0.2883 & 0.3047 & 0.6650 \\ 0.4047 & 0.8828 & 0.8732 & 0.5743 & 0.1091 & 0.0381 \end{pmatrix}$$

the value of global minimum is -3.322368.

- **Rastrigin** function. The function is given by

$$f(x) = x_1^2 + x_2^2 - \cos(18x_1) - \cos(18x_2), \quad x \in [-1,1]^2$$

The global minimum is located at $x^* = (0,0)$ with value -2.0.

- ***Rosenbrock*** *function*
This function is given by

$$f(x) = \sum_{i=1}^{n-1}\left(100\left(x_{i+1} - x_i^2\right)^2 + \left(x_i - 1\right)^2\right), \quad -30 \le x_i \le 30.$$

The global minimum is located at the $x^* = (0,0,...,0)$ with $f\left(x^*\right) = 0$. In our experiments we used this function with $n = 20$.

- **Shekel 7** function.

$$f(x) = -\sum_{i=1}^{7} \frac{1}{(x-a_i)(x-a_i)^T + c_i}$$

with $x \in [0,10]^4$ and $a = \begin{pmatrix} 4 & 4 & 4 & 4 \\ 1 & 1 & 1 & 1 \\ 8 & 8 & 8 & 8 \\ 6 & 6 & 6 & 6 \\ 3 & 7 & 3 & 7 \\ 2 & 9 & 2 & 9 \\ 5 & 3 & 5 & 3 \end{pmatrix}$, $c = \begin{pmatrix} 0.1 \\ 0.2 \\ 0.2 \\ 0.4 \\ 0.4 \\ 0.6 \\ 0.3 \end{pmatrix}$. The value of global minimum is -10.342378.

- **Shekel 5** function.

$$f(x) = -\sum_{i=1}^{5} \frac{1}{(x-a_i)(x-a_i)^T + c_i}$$

with $x \in [0,10]^4$ and $a = \begin{pmatrix} 4 & 4 & 4 & 4 \\ 1 & 1 & 1 & 1 \\ 8 & 8 & 8 & 8 \\ 6 & 6 & 6 & 6 \\ 3 & 7 & 3 & 7 \end{pmatrix}$, $c = \begin{pmatrix} 0.1 \\ 0.2 \\ 0.2 \\ 0.4 \\ 0.4 \end{pmatrix}$. The value of global minimum is -10.107749.

- **Shekel 10** function.

$$f(x) = -\sum_{i=1}^{10} \frac{1}{(x-a_i)(x-a_i)^T + c_i}$$

with $x \in [0,10]^4$ and $a = \begin{pmatrix} 4 & 4 & 4 & 4 \\ 1 & 1 & 1 & 1 \\ 8 & 8 & 8 & 8 \\ 6 & 6 & 6 & 6 \\ 3 & 7 & 3 & 7 \\ 2 & 9 & 2 & 9 \\ 5 & 5 & 3 & 3 \\ 8 & 1 & 8 & 1 \\ 6 & 2 & 6 & 2 \\ 7 & 3.6 & 7 & 3.6 \end{pmatrix}$, $c = \begin{pmatrix} 0.1 \\ 0.2 \\ 0.2 \\ 0.4 \\ 0.4 \\ 0.6 \\ 0.3 \\ 0.7 \\ 0.5 \\ 0.6 \end{pmatrix}$. The value of global minimum is -10.536410.

- **Sinusoidal** function. The function is given by

$$f(x) = -\left(2.5 \prod_{i=1}^{n} \sin\left(x_i - z\right) + \prod_{i=1}^{n} \sin\left(5\left(x_i - z\right)\right)\right), \quad 0 \le x_i \le \pi.$$

The global minimum is located at $x^* = (2.09435, 2.09435, ..., 2.09435)$ with $f(x^*) = -3.5$. In our experiments we used $n = 4, 8, 16, 32$ and $z = \frac{\pi}{6}$ and the corresponding functions are denoted by the labels SINU4, SINU8, SINU16, SINU32.

- **Test2N** function. This function is given by the equation

$$f(x) = \frac{1}{2}\sum_{i=1}^{n} x_i^4 - 16x_i^2 + 5x_i, \quad x_i \in [-5, 5].$$

The function has 2^n in the specified range and in our experiments we used $n = 4, 5, 6, 7$. The corresponding values of global minimum is -156.664663 for $n = 4$, -195.830829 for $n = 5$, -234.996994 for $n = 6$ and -274.163160 for $n = 7$.

- **Test30N** function. This function is given by

$$f(x) = \frac{1}{10}\sin^2(3\pi x_1)\sum_{i=2}^{n-1}\left((x_i-1)^2\left(1+\sin^2(3\pi x_{i+1})\right)\right)+(x_n-1)^2\left(1+\sin^2(2\pi x_n)\right)$$

with $x \in [-10, 10]$. The function has 30^n local minima in the specified range and we used $n = 3, 4$ in our experiments. The value of global minimum for this function is 0.0

3.2 Results

4 Conclusions

References

[1] Törn A. and Žilinskas A., Global Optimization Volume 350 of Lecture Notes in Computer Science, Springer, Heidelberg, 1987.

[2] Patrice Ogou Yapo, Hoshin Vijai Gupta, Soroosh Sorooshian, Multi-objective global optimization for hydrologic models, Journal of Hydrology 204, pp. 83-97, 1998.

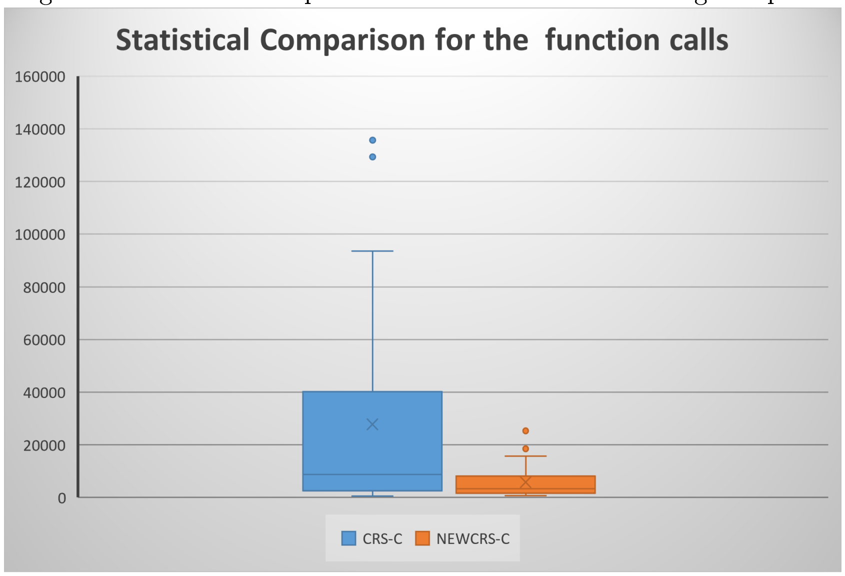
[3] Q. Duan, S. Sorooshian, V. Gupta, Effective and efficient global optimization for conceptual rainfall-runoff models, Water Resources Research **28**, pp. 1015-1031 , 1992.

[4] Balsa-Canto E., Banga J.R., Egea J.A., Fernandez-Villaverde A., de Hijas-Liste G.M. (2012) Global Optimization in Systems Biology: Stochastic Methods and Their Applications. In: Goryanin I., Goryachev A. (eds) Advances in Systems Biology. Advances in Experimental Medicine and Biology, vol 736. Springer, New York, NY. https://doi.org/10.1007/978-1-4419-7210-1_24

Table 1: Experimenting with rejection rates.

FUNCTION	CRS-R	CRS-C	NEWCRS-R	NEWCRS-C
BF1	1.37%	2523	0.00%	1689
BF2	1.33%	2506	0.17%	1569
BRANIN	16.00%	2014	9.13%	851
CAMEL	1.67%	2235	0.20%	1487
EASOM	51.03%	591	11.43%	635
EXP2	3.03%	1290	0.70%	644
EXP4	2.67%	4688	0.00%	1302
EXP8	2.77%	16453	0.00%	2601
EXP16	4.00%	47400	0.00%	5207
EXP32	7.70%	93520	0.00%	10414
EXP64	18.80%	135638	0.00%	13602
EXP100	38.53%	129327	0.00%	14506
GKLS250	3.87%	1784	0.27%	1684
GKLS350	6.43%	3881	0.03%	2088
GOLDSTEIN	3.60%	2154	0.70%	1829
GRIEWANK2	1.20%	2503	0.03%	2742
GUILIN550	8.33%	9129	0.00%	25333
GUILIN1050	9.63%	30806	0.00	10561
HANSEN	47.60%	2643	4.03%	1736
HARTMAN3	9.97%	3009	6.13%	1331
HARTMAN6	13.37%	13615	0.00%	6091
RASTRIGIN	9.17%	2130	1.33%	2986
ROSENBROCK	0.00%	59024	0.00%	15719
SHEKEL5	4.73%	8974	0.00%	2967
SHEKEL7	3.70%	8606	0.00%	3236
SHEKEL10	2.73%	9264	0.00%	3479
SINU4	3.90%	6525	0.00%	2889
SINU8	5.10%	21561	0.00%	4946
SINU16	8.43%	62194	0.00%	9539
SINU32	14.40%	135986	0.00%	18456
TEST2N4	24.57%	10198	0.00%	3756
TEST2N5	34.17%	20850	0.00%	4806
TEST2N6	42.50%	43290	0.00%	6075
TEST2N7	50.37%	92658	0.00%	7005
TEST30N3	24.10%	4011	0.00%	5691
TEST30N4	27.30%	7432	0.00%	8579
TOTAL	**13.67%**	1000412	**0.86%**	208031

Figure 1: Statistical comparison for the function calls using box plots.



[5] P. Boutros, A. Ewing, K. Ellrott et al., Global optimization of somatic variant identification in cancer genomes with a global community challenge, Nat Genet **46**, pp. 318–319, 2014.

[6] David J. Wales, Harold A. Scheraga, Global Optimization of Clusters, Crystals, and Biomolecules, Science **27**, pp. 1368-1372, 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] 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.

[9] S. Kirkpatrick, CD Gelatt, , MP Vecchi, Optimization by simulated annealing, Science **220**, pp. 671-680, 1983.

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

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

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

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

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

[15] A. Duarte, R. Martí, F. Glover et al., Hybrid scatter tabu search for unconstrained global optimization, Ann Oper Res **183**, pp. 95–123m, 2011.

[16] J. Kennedy and R. Eberhart, "Particle swarm optimization," Proceedings of ICNN'95 - International Conference on Neural Networks, 1995, pp. 1942-1948 vol.4, doi: 10.1109/ICNN.1995.488968.

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

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

[19] W.L. Price, Global Optimization by Controlled Random Search, Computer Journal **20**, pp. 367-370, 1977.

[20] David N. Smith and John F. Ferguson, Constrained inversion of seismic refraction data using the controlled random search, Geophysics **65**,pp. 1622-1630, 200.

[21] Cassiano Antonio Bortolozo, Jorge Luís Porsani, Fernando Acácio Monteiro dos Santos, Emerson Rodrigo Almeida, VES/TEM 1D joint inversion by using Controlled Random Search (CRS) algorithm, Journal of Applied Geophysics **112**, pp. 157-174, 2015.

[22] J. Haslinger, D. Jedelský, T. Kozubek et al, Genetic and Random Search Methods in Optimal Shape Design Problems, Journal of Global Optimization **16**, pp. 109–131, 2000.

[23] Radha Gupta, Manasa Chandan, Use of "Controlled Random Search Technique for Global Optimization" in Animal Diet Problem, International Journal of Emerging Technology and Advanced Engineering **3**, pp. 284-287, 2013.

[24] R.C. Mehta, S.B. Tiwari, Controlled random search technique for estimation of convective heat transfer coefficient, Heat Mass Transfer **43**, pp.1171–1177, 2007.

[25] M.M. Ali, C. Storey, Modified Controlled Random Search Algorithms, International Journal of Computer Mathematics **53**, pp. 229-235, 1994.

[26] Di Pillo G., Lucidi S., Palagi L., Roma M. (1998) A Controlled Random Search Algorithm with Local Newton-type Search for Global Optimization. In: De Leone R., Murli A., Pardalos P.M., Toraldo G. (eds) High Performance Algorithms and Software in Nonlinear Optimization. Applied Optimization, vol 24. Springer, Boston, MA. https://doi.org/10.1007/978-1-4613-3279-4_10

[27] S. Lucidi, F. Rochetich, and M. Roma, Curvilinear stabilization techniques for truncated Newton methods in large scale unconstrained optimization, SIAM Journal on Optimization **8**, pp. 916–939, 1998.

[28] P. Kaelo, M.M. Ali, Some Variants of the Controlled Random Search Algorithm for Global Optimization, Journal of Optimization Theory and Applications **130**, pp. 253–264, 2006.

[29] Nelson Manzanares-filho , Rodrigo B. F. Albuquerque, Accelerating Controlled Random Search Algorithms Using a Distribution Strategy, In: EngOpt 2008 - International Conference on Engineering Optimization, Rio de Janeiro, Brazil, 01 - 05 June 2008.

[30] Ioannis G. Tsoulos, Modifications of real code genetic algorithm for global optimization, Applied Mathematics and Computation **203**, pp. 598-607, 2008.

[31] 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.

[32] 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.

[33] M. Gaviano, D.E. Ksasov, D. Lera, Y.D. Sergeyev, Software for generation of classes of test functions with known local and global minima for global optimization, ACM Trans. Math. Softw. **29**, pp. 469-480, 2003.