 doi:10.20944/preprints202004.0273.v1

Cooperative Multi-Simplex Algorithm: An Innovation from Localization to Globalization

Javaid Ali

Department of Mathematics, University of Management and Technology, Lahore, Pakistan

javaidaliwaseer@gmail.com

Abstract

This study suggests a novel cooperative multi-simplex algorithm that generalizes a local search optimizer to design a novel global search heuristic algorithm. The proposed algorithm exploits the vertex sharing strategy to enhance the search abilities of the working simplexes. The vertex sharing among the simplexes is carried out through cooperative step that is based on fitness of the underlying simplex. The proposed algorithm is applied to solve some systems of nonlinear equations by transforming them to optimization problems. Comparative analysis of results shows that the proposed method is practical and effective.

Keywords: Nelder-Mead algorithm; cooperative multi-simplex algorithm; simplex-fitness; system of nonlinear equations

1. Introduction

Nelder-Mead Simplex (NMS) algorithm [1] is a classical method for numerical optimization of unconstrained problems. If $n \in \mathbb{Z}^{+}$ then for solving n dimensional problem NMS method uses a convex hull of $n+1$ points, usually called a simplex. The method involves four steps, namely, (i) reflection (ii) expansion (iii) contraction and (iv) shrinkage with the help of scalars $\alpha = 1,\ \beta = 2, \gamma = 0.5$ and $\delta = -0.5$ [1, 2, 3].

Consider $\boldsymbol{V}_j \in \mathbb{R}^n; 1 \le j \le n+1$ be the vertices of the Polytopes with corresponding function values f_j arranged in ascending order $f_j \le f_l\ \forall j \le l$. The NMS method calculates the centroid $\boldsymbol{G}$ by relation (4) and then uses (5)-(8) to improve $\boldsymbol{V}_{n+1}$ by generating points $\boldsymbol{R}, \boldsymbol{E}, \boldsymbol{C}^{out}$ or $\boldsymbol{C}^{in}$.

$$\boldsymbol{G} = \frac{1}{n}\sum_{j=1}^{n} \boldsymbol{V}_j \qquad (4)$$

$$\boldsymbol{R} = \boldsymbol{G} + 1 \times (\boldsymbol{G} - \boldsymbol{V}_{n+1}) \qquad (5)$$

$$\boldsymbol{E} = \boldsymbol{G} + 2 \times (\boldsymbol{G} - \boldsymbol{V}_{n+1}) \qquad (6)$$

$$\boldsymbol{C}^{out} = \boldsymbol{G} + 0.5 \times (\boldsymbol{G} - \boldsymbol{V}_{n+1}) \qquad (7)$$

$$\boldsymbol{C}^{in} = \boldsymbol{G} - 0.5 \times (\boldsymbol{G} - \boldsymbol{V}_{n+1}) \qquad (8)$$

 doi:10.20944/preprints202004.0273.v1

The fifth operation is the shrink step that comes into action when points generated by (5)-(8) fail to improve $\boldsymbol{V}_{n+1}$ [2]. Figure 1 shows the geometry of the operations of NMS method in $\mathbb{R}^2$ [3].

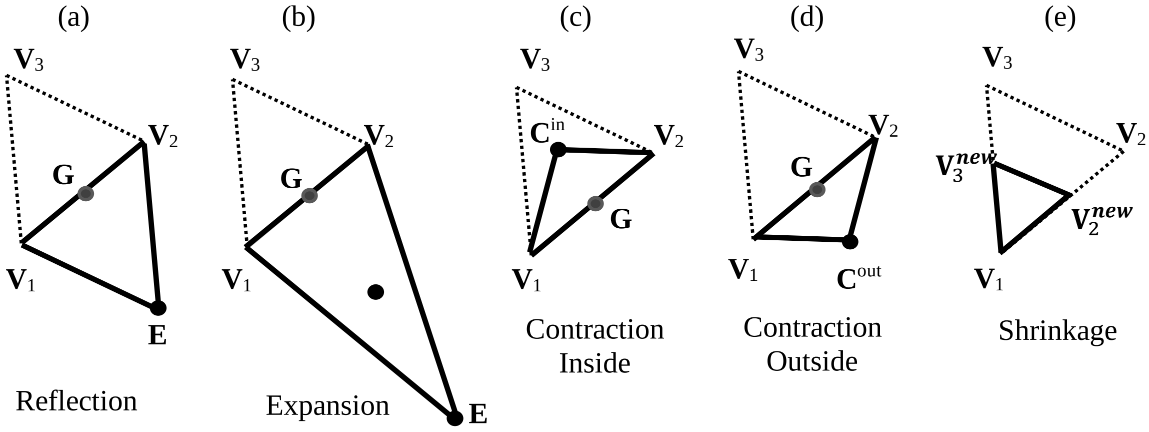

Figure 1. Operations on a simplex in $\mathbb{R}^2$

A general iteration of original NMS method in $\mathbb{R}^n$ is restated as under [1, 2, 4].

An Iteration of NMS method:

1. *Ordering:* Arrange vertices as $f_j \le f_l \ \forall \ j \le l$.
2. *Reflection*: Compute $\boldsymbol{R}$, if $f(\boldsymbol{R}) \in [f_1, f_{n+1})$, save $\boldsymbol{R}$ as $\boldsymbol{V}^{new}$.
3. *Expansion*: If $f(\boldsymbol{R}) < f_1$ compute $\boldsymbol{E}$, if $f(\boldsymbol{E}) < f(\boldsymbol{R})$ save $\boldsymbol{E}$ as $\boldsymbol{V}^{new}$.
4. *Contraction Outside*: If $f(\boldsymbol{R}) \in [f_n, f_{n+1})$ find $\boldsymbol{C}^{out}$, if $f(\boldsymbol{C}^{out}) \le f(\boldsymbol{R})$ save $\boldsymbol{C}^{out}$ as $\boldsymbol{V}^{new}$.
5. *Contraction Inside*: If $f(\boldsymbol{R}) \ge f_{n+1}$ determine $\boldsymbol{C}^{in}$, **if** $f\left(\boldsymbol{C}^{in}\right) < f_{n+1}$ store $\boldsymbol{C}^{in}$ as $\boldsymbol{V}^{new}$.
6. *Shrinkage*: If $f(\boldsymbol{V}^{new}) < f_{n+1}$ then set $\boldsymbol{V}^{new}$ to $\boldsymbol{V}_{n+1}$ otherwise execute shrinkage step:
   $\boldsymbol{V}_j \leftarrow \boldsymbol{V}_j + 0.5 \times \left(\boldsymbol{V}_1 - \boldsymbol{V}_j\right) \ \forall \ j \in \{2, 3, 4, \ldots, n+1\}$.

2. Related works on the proposed Cooperative Multi-Simplex (CMS) algorithm

The proposed cooperative multi-simplex algorithm (CMS) algorithm starts by randomly generating N_s, $N_s \in \mathbb{Z}^+$, simplexes in the search space. The iterative process of the proposed CMS algorithm is comprised of a cooperative step and a rotational shrinkage based modified iteration of NMS method. The cooperative step establishes a probability based sharing among the vertices of various simplexes. Based upon a user-defined cooperative sharing probability $p \in [0, 1]$, the vertex sharing is divided in to mixed sharing and ascent sharing.

To elaborate more clearly, suppose $S^{(i,k)}$ is the set of vertices belonging to i^{th} simplex $i \in [1, N_s]$ at k^{th} iteration, the centroid $\boldsymbol{G}^{(i,k)}$, calculated by relation (4) relates to $S^{(i,k)}$ and let

$f_j^{(i,k)} = f\left(\boldsymbol{V}_j^{(i,k)}\right); 1 \leq j \leq n+1$. With these notations, the main steps of CMS algorithm are summarized as under.

3.1. *Initialization*

Generate N_s simplexes $S^{(i,k)}: 1 \leq i \leq N_s$, choose a suitable value of p and set an integer FE_{max} as maximum number of function evaluations allowed.

3.2. *Ordering*

Sort all the vertices of the each simplex:

$$f_1^{(i,k)} \leq f_2^{(i,k)} \leq f_3^{(i,k)} \leq \cdots \leq f_{n+1}^{(i,k)} \qquad (9)$$

3.3. *Cooperative step*

The attribute of cooperative sharing and exploiting the information composed from the entire population are crucial tools of population based heuristic algorithms [5, 6, 7] which empower them to perform balanced exploration and exploitation in optimization process. In CMS algorithm, the cooperative step handles the sharing of vertices based on the fitness of the simplexes. It not only alters orientations of the corresponding simplexes but also enforces them to cluster around the promising locations in the search space. The fitness of a simplex is calculated by using Equations (10) and (11) in turn.

$$Fit^{(i,k)} = \frac{1}{1+\underline{f}^{(i,k)}} \qquad (10)$$

$$\underline{f}^{(i,k)} = \frac{1}{(n+1)} \sum_{j=1}^{n+1} \left(\frac{1}{1+f_j^{(i,k)}}\right) \qquad (11)$$

Two real numbers μ and $\lambda \in [0,1]$ are generated randomly. The sharing of a vertex of some i^{th} simplex with another simplex takes place if $\lambda > Fit^{(i,k)}$, otherwise letting the simplexes proceed independently. If $\mu < p$, the mixed sharing exchanges the non-best vertex of a randomly simplex with some non-best vertex of the current simplex whereas the ascent sharing replaces the worst vertex of the current simplex by the worst vertex of some other simplex otherwise.

3.4. *Rotational shrinkage based iteration of NMS method*

The proposed CMS heuristic method executes the standard operations of reflection, expansion and contraction but a different shrinkage step, called rotational shrinkage. The proposed rotational shrinkage step aims to change the orientation of the current simplex and to increase

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 16 April 2020 doi:10.20944/preprints202004.0273.v1

the exploration chances without utilizing additional computational cost. The rotational shrinkage generates new vertices as follows.

$$\boldsymbol{V}_j^{new} = \boldsymbol{V}_1^{(i,k)} + \delta\left(\boldsymbol{V}_1^{(i,k)} - \boldsymbol{V}_j^{(i,k)}\right) \text{ for } j = 2, 3, 4, \dots, n+1 \quad (12)$$

The new simplex for the $(k+1)^{th}$ iteration is constructed using the following conditions.

$$S^{(i,k+1)} = \begin{cases} (S^{(i,k)} \setminus \{\boldsymbol{V}_{n+1}^{(i,k)}\}) \cup \{\boldsymbol{V}^{new}\} & \text{if no shrinkage occurs} \\ \left\{\boldsymbol{V}_j^{new} : 2 \le j \le n+1\right\} \cup \left\{\boldsymbol{V}_1^{(i,k)}\right\} & \text{if shrinkage takes place} \end{cases} \quad (13)$$

During the iterative process, the best of all of the vertices of N_s simplexes is retained and is updated at each function evaluation. The iterative process of CMS method continues up to a predefined budget (FE_{max}) of function evaluations. The Algorithm 2 presents the pseudo code of the proposed CMS method.

Algorithm 2: Pseudo code of the proposed CMS algorithm

INITIALIZE: Generate N_s simplexes; define cooperative probability p and the budget FE_{max}; set the function evaluations counter: $NFEs = 0$; set $k = 1$. Retain the best of $N(n+1)$ vertices as the current best solution.

{WHILE *(NFEs< FEmax)*

  {FOR $i = 1, 2, 3, \dots, N_s$

    Order $S^{(i,k)}$ to satisfy Equation (9), calculate $F^{(i,k)}$ using Equations (10) and (11). Choose $\lambda \in [0,1]$ randomly.

    {IF $\lambda > F^{(i,k)}$ choose $\mu \in [0,1]$ randomly

      {IF $\mu < p$ apply mixed sharing on $S^{(i,k)}$

      ELSE apply ascent sharing on $S^{(i,k)}$

      ENDIF}

      Order $S^{(i,k)}$ to satisfy sequence (9).

    ENDIF}

    Apply rotational shrinkage based NMS-iteration on $S^{(i,k)}$.

  ENDFOR}

  Update *NFEs* and the best solution. Set $k = k + 1$.

ENDWHILE}

---

3. Applications of proposed CMS algorithm to physical systems and numerical results

In order to validate the effectiveness of our proposed CMS algorithm, two mathematical and one physical system of non-linear equations are utilized. Four state of the art algorithms, namely,

 doi:10.20944/preprints202004.0273.v1

Particle Swarm Optimization (PSO) [5], Differential Evolution (DE) [6], Artificial Bee Colony (ABC) [8] and Teaching Learning Based Optimization (TLBO) [9] are considered for the performance comparisons.

4.1. *Mathematical test problem 1*

The first test problem has been taken from [10-13]. This problem is described by the system (14) of non-linear equations.

$$\begin{cases} E_1(\boldsymbol{x}) = x_1^{x_2} + x_2^{x_1} - 5\,x_1x_2 - 85 = 0, \\ E_2(\boldsymbol{x}) = x_1^3 - x_2^{x_3} - x_3^{x_2} - 60 = 0, \\ E_3(\boldsymbol{x}) = x_1^3 + x_3^{x_1} - x_2 - 62 = 0, \\ 3 \le x_1 \le 5,\ 2 \le x_2 \le 4,\ 0.5 \le x_3 \le 2. \end{cases} \tag{14}$$

The exact solution to the system reported is (4, 3, 1).

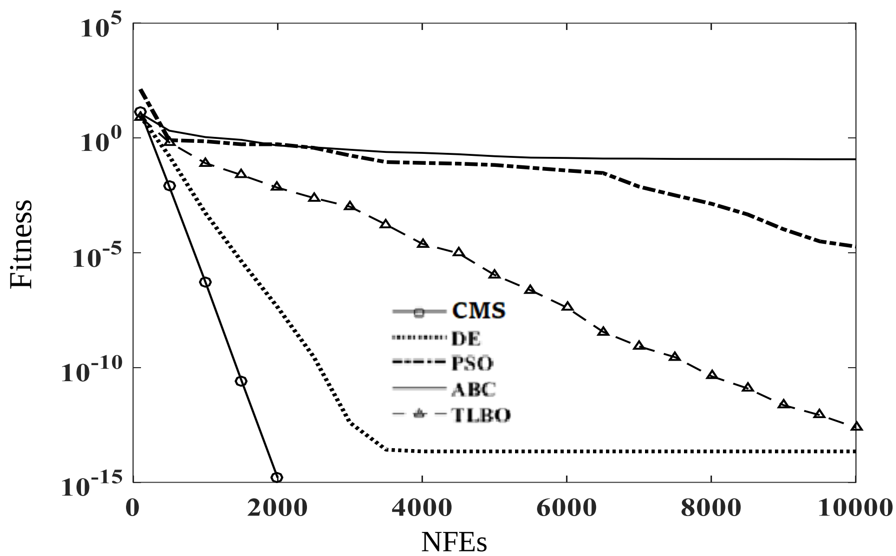

Figure 2. Convergence curves for mathematical test problem 1.

Table 1. Numerical results on mathematical test problem 1.

	Proposed CMS	DE	PSO	ABC	TLBO
x_1	**4**	**4**	4	4.000526	4
x_2	**3**	**3**	3	3.000485	3
x_3	**1**	**1**	0.999999	1.002746	1.000001
B	**0**	**0**	1.42E-06	1.32E-02	1.28E-14
Md	**0**	3.34E-14	1.71E-05	1.71E-01	2.57E-13
Mn	**9.2E-16**	7.36 E-03	1.39 E-01	1.97E-01	8.83E-13

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 16 April 2020 doi:10.20944/preprints202004.0273.v1

4.2. *Mathematical test problem 2*

The second problem has been extracted from [14] and involves following four equations:

$$\begin{cases} E_1(\boldsymbol{x}) = x_2x_3 + (x_2 + x_3)x_4 = 0, \\ E_2(\boldsymbol{x}) = x_1x_3 + (x_1 + x_3)x_4 = 0, \\ E_3(\boldsymbol{x}) = x_1x_2 + (x_1 + x_2)x_4 = 0, \\ E_4(\boldsymbol{x}) = x_1x_2 + x_1x_3 + x_2x_3 - 1 = 0, \\ \quad -1 \le x_1, x_2, x_3, x_4 \le 1 \end{cases} \tag{15}$$

Table 2. Numerical results on mathematical test problem 2.

	Proposed CMS	DE	PSO	ABC	TLBO
x_1	**0.5773502692**	**-0.5773502692**	0.57736	0.554	-0.5773502691
x_2	**0.5773502692**	**-0.5773502692**	0.57734	0.582	-0.57735027
x_3	**0.5773502692**	**-0.5773502692**	0.577348	0.598	-0.577350268
x_4	**-0.28867513459**	**0.28867513459**	-0.288675	-0.2887	0.28867513452
B	**0**	**0**	1.27E-06	2.37E-03	1.86E-19
Md	**0**	**0**	2.41E-05	5.73E-03	1.09E-08
Mn	**5.44E-18**	2.71E-08	4.95E-03	5.77E-03	7.09E-06

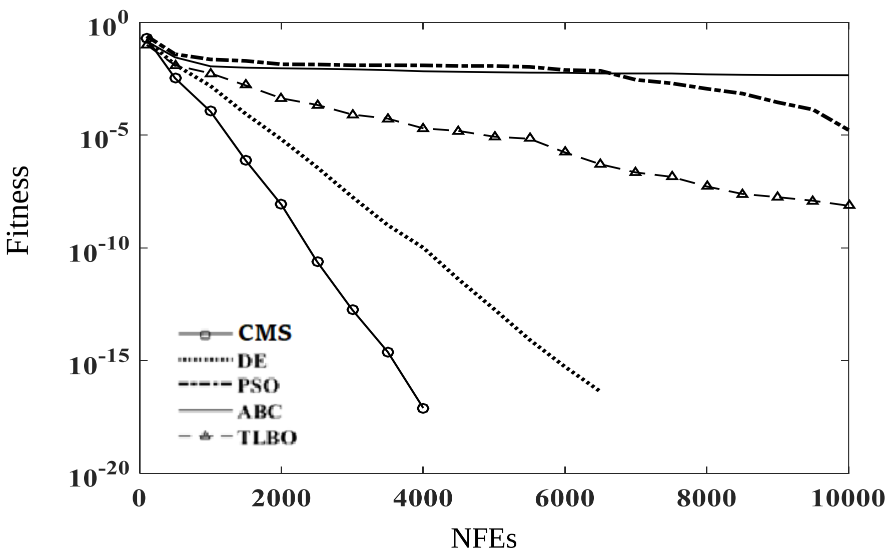

Figure 3. Convergence curves for mathematical test problem 2

4.3. *Thin wall rectangle girder section problem*

Geometry size of thin wall rectangle girder section problem involves following system of equations [10, 12, 15, 16].

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 16 April 2020 doi:10.20944/preprints202004.0273.v1

$$\begin{cases} E_1(\boldsymbol{x}) = x_1x_2 - (x_2 - 2x_3)(x_1 - 2x_3) - 165 = 0, \\ E_2(\boldsymbol{x}) = \frac{x_1^3x_2}{12} - \frac{(x_2-2x_3)(x_1-2x_3)^3}{12} - 9369 = 0, \\ E_3(\boldsymbol{x}) = 2x_3(x_1 - x_3)^2(x_2 - x_3)^2/(x_1 + x_2 - 2x_3) - 6835 = 0. \end{cases} \quad (16)$$

Where x_1, x_2 and x_3 are height, width and thickness of the section respectively. The physical constraints on the system are:

$$g_1(\boldsymbol{x}) = x_3 > 0;\ g_2(\boldsymbol{x}) = x_2 - x_3 > 0;\ g_2(\boldsymbol{x}) = x_1 - x_2 > 0. \quad (17)$$

Table 3. Numerical results on girder section problem.

	Proposed CMS	DE	PSO	ABC	TLBO
x_1	**22.89494**	22.892	22.95	23.29	22.89622
x_2	**12.25652**	12.2564	12.258	12.28	12.25653
x_3	**2.789818**	2.7916	2.76	2.57	2.78904
B	**3.79E-22**	3.87E-02	4.40E-02	3.18	1.69E-02
Md	**3.03E-13**	13.50	10.00	14.72	5.05
Mn	12.321	14.46	13.85	16.28	**9.72**

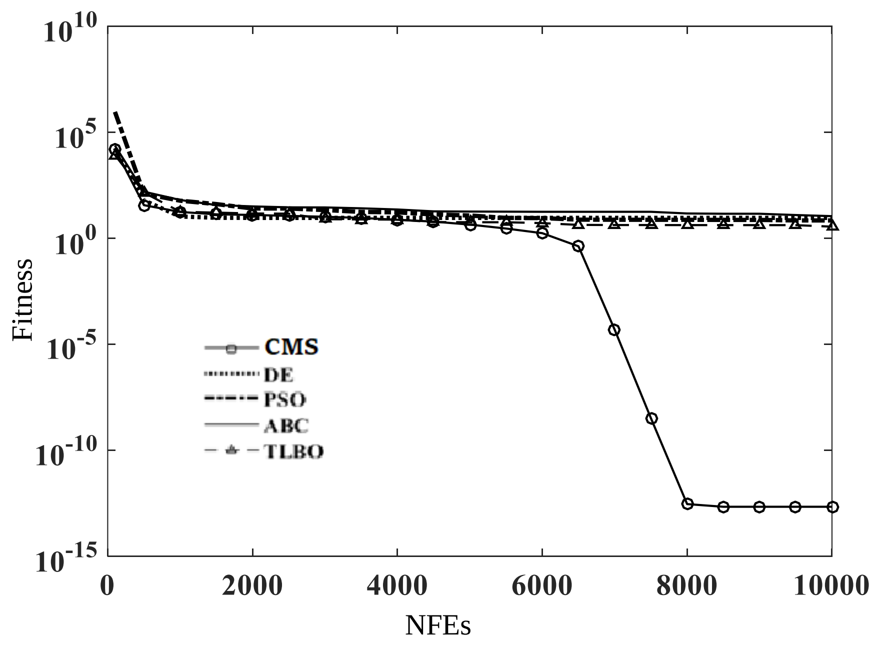

Figure 4. Convergence curves for mathematical test thin wall girder section problem.

4. Conclusion

This study presents a novel approach for solving a system of nonlinear equations as an optimization problem. The proposed method neither requires initial guess nor derivative information. The analysis has been conducted through detailed and logical comparisons based on statistical measures the Best values (*B*), Median values (*Md*) and Mean values (*Mn*). It can be observed from Tables 1-3 that the solutions produced by the proposed CMS algorithm are more accurate solutions (*B*, *Md*, *Mn*). The convergence graphs shown in Figures 2-4 evidently demonstrates that the developed CMS algorithm significantly outperforms DE, PSO, ABC and TLBO in terms of solution quality and convergence speed.

The proposed work can be extended to several disciplines of numerical optimization in collaboration with general purpose global search optimization algorithms.

References

[1] Nelder, J. A., and R. Mead. 1965. "A Simplex Method for Function Minimization." *The Computer Journal* 7(4): 308-313. doi: 10.1093/comjnl/7.4.308.

[2] Price, C. J., I. D. Coope, and D. Byatt. 2002. "A Convergent Variant of the Nelder-Mead Algorith." *Journal of Optimization Thoery and Application* 113 (1): 5-19. doi: 10.1023/A:1014849028575.

[3] Lagarias, J. C., B. Poonen, and M. H. Wright. 2012. "Convergence of the Restricted Nelder–Mead Algorithm in Two Dimensions." *SIAM Journal of Optimization* 22(2):501–532. doi: 10.1103/PhysRevLett.81.1195.

[4] Ali, J., M. Saeed, N. A. Chaudhry, M. F. Tabassum, and M. Luqman. 2017. "Low Cost Efficient Remedial Strategy for Stagnated Nelder-Mead Simplex Method." *Pakistan Journal of Science* 69(1): 119–126.

[5] Kennedy, J., and R. C. Eberhart. 1995. "Particle Swarm Optimization." *Proceedings of the IEEE International Conference on Neural Networks,* Perth, WA, Australia, 27 Nov. - 1 Dec. 1995: 1942–1948. New York: IEEE.

[6] Storn, R., and K. Price. 1997. "Differential Evolution: A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces." *Journal of Global Optimization* 11(4):341-359. doi: 10.1023/A:1008202821328.

[7] Ali, J., M. Saeed, M. F. Tabassam, and S. Iqbal. 2019. "Controlled Showering Optimization Algorithm: An Intelligent Tool for Decision Making in Global Optimization." *Computational Mathematical Organizational Theory* 25**,** 132–164. doi: https://doi.org/10.1007/s10588-019-09293-6

[8] Karaboga, D., and B. Basturk. 2007. "A Powerful and Efficient Algorithm for Numerical Function Optimization: Artificial Bee Colony Algorithm." *Journal of Global Optimization* 39(3): 459–471. doi: 10.1007/s10898-007-9149-x.

[9] Rao, R. V., V. J. Savsani, and D. P. Vakharia. 2011. "Teaching Learning Based Optimization: A Novel Method for Constrained Mechanical Design Optimization Problems." *Computer-aided Design* 43 (3): 303-315. doi: 10.1016/j.cad.2010.12.015.

[10] Abdollahi, M., A. Isazadeh, and D. Abdollahi. 2013. "Imperialist Competitive Algorithm for Solving Systems of Nonlinear Equations." *Computers and Mathematics with Applications* 65(12): 1894–1908. doi: 10.1016/j.camwa.2013.04.018.

[11] Jaberipour, M., E. Khorram, and B. Karimi. 2011. "Particle Swarm Algorithm for Solving Systems of Nonlinear Equations." *Computers and Mathematics with Applications* 62(2): 566–576. doi: 10.1016/j.camwa.2011.05.031.

[12] Mo, Y., H. Liu, and Q. Wang. 2009. "Conjugate Direction Particle Swarm Optimization Solving Systems of Nonlinear Equations." *Computers and Mathematics with Applications* 57(11-12): 1877–1882. doi: 10.1016/j.camwa.2008.10.005.

[13] Luo, Y. Z., G. J. Tang, and L. N. Zhou. 2008. "Hybrid Approach for Solving Systems of Nonlinear Equations Using Chaos Optimization and Quasi-Newton Method." *Applied Soft Computing* 8(2): 1068–1073. doi: 10.1016/j.asoc.2007.05.013.

[14] Ahmad, F., E. Tohidi, M. Z. Ullah, and J. A. Carrasco. 2015. "Higher Order Multi-Step Jarratt-Like Method for Solving Systems of Nonlinear Equations: Application to PDEs and ODEs." *Computers & Mathematics with Applications* 70(4): 624-636. doi: 10.1016/j.camwa.2015.05.012.

[15] Morgan, A. P. 1987. "Computing All Solutions to Polynomial Systems Using Homotopy Continuation." *Applied Mathematics and Computation* 24(2): 115–138. doi: 10.1016/0096-3003%2887%2990064-6.

[16] Luqman, M., M. Saeed, and J. Ali. 2017. "On Solution of Nonlinear Models: A Hybrid Algorithm of Artificial Bee Colony Algorithm and Artificial Showering Algorithm." *Pakistan Journal of Statistics* 33(5):399-409.