Swarm and Evolutionary Computation 48 (2019) 72–92

Contents lists available at ScienceDirect

Swarm and Evolutionary Computation

journal homepage: www.elsevier.com/locate/swevo

Solving systems of nonlinear equations using a modified firefly algorithm (MODFA)

M.K.A. Ariyaratne [a,*], T.G.I. Fernando [a], S. Weerakoon [b]

[a] Department of Computer Science, Faculty of Applied Sciences, University of Sri Jayewardenepura, Sri Lanka
[b] Department of Mathematics, Faculty of Applied Sciences, University of Sri Jayewardenepura, Sri Lanka

ARTICLE INFO

Keywords:
Firefly algorithm
Systems of nonlinear equations
Swarm intelligence
Functions of several variables

ABSTRACT

Most numerical methods that are being used to solve systems of nonlinear equations require the differentiability of the functions and acceptable initial guesses. However, some optimization techniques have overcome these problems, but they are unable to provide more than one root approximation simultaneously. In this paper, we present a modified firefly algorithm treating the problem as an optimization problem, which is capable of giving multiple root approximations simultaneously within a reasonable state space. The new method does not concern initial guesses, differentiability and even the continuity of the functions. Results obtained are encouraging, giving sufficient evidence that the algorithm works well. We further illustrate the viability of our method using benchmark systems found in the literature.

1. Introduction

One of the most explored problems in applied mathematics, engineering and sciences is finding multiple solutions of a set of nonlinear equations [1,2]. For an example, the book titled "Nonlinear Optimization with Engineering applications" [3] reveals a comprehensive study of problems in Science and Engineering such as data fitting, vehicle route planning, scheduling and resource allocating where the involvement of nonlinear systems is high. When a system is linear, solutions can be obtained easily. But nonlinear systems are not that easily solvable. Moreover, when the number of equations increases the difficulties are magnified. Despite its complexity, there is a number of approaches including numerical methods and optimization techniques used over the years to solve systems of nonlinear equations [4–6]. Many famous techniques to solve nonlinear equations are based on the Newton's method [7]. But the need of good initial guesses, continuity and the differentiability of the functions limit the usability of those techniques. Optimization techniques such as Particle Swarm Optimization (PSO), Genetic Algorithms (GA), Harmony Search (HS) and heuristics like continuous greedy randomized adaptive search procedures (CGRASP) were also used to solve systems of nonlinear equations [6,8–11]. In such situations, the systems of nonlinear equations were treated as a single objective or multi objective optimization problem depending on the circumstance. These optimization techniques are simpler than numerical approaches as the derivative information is not used and hence large matrices are not present. However they are not so successful in solving the problem completely.

In this paper we present an optimization algorithm to solve a given system of nonlinear equations using the firefly algorithm (FA) [12]. The original FA has been modified to accomplish the purpose by finding all known root approximations simultaneously within a given state space. Yang has implemented the original FA to find a single global optimum [13]. Since a system of nonlinear equations normally have multiple roots, it possesses more than one global optimum. Because of that we have modified the original FA to produce multiple optimal solutions.

The remainder of this paper is structured as follows. Section two is dedicated for the related work. Section three describes the preliminaries with respect to the systems of nonlinear equations and the basic firefly algorithm. Modified FA (MODFA) and how it can be applied to solve systems of nonlinear equations are explained in section four. Section five presents numerical examples used and section six presents the results obtained by the new algorithm. Finally, section seven concludes and points out future directions possible.

* Corresponding author.
E-mail addresses: [illegible]radhaariyaratne@gmail.com (M.K.A. Ariyaratne), gishantha@dscs.sjp.ac.lk (T.G.I. Fernando), sunethra.weerakoon@gmail.com (S. Weerakoon).

https://doi.org/10.1016/j.swevo.2019.03.010
Received 18 July 2017; Received in revised form 19 March 2019; Accepted 20 March 2019
Available online 29 March 2019

2. Related work

Numerical methods are the first and the most used methods in earlier days to solve nonlinear equations as well as systems. For single variable nonlinear equations there are many numerical methods such as Newton's method, Secant method and Bisection method. These methods work well when good initial guesses are provided and when the functions are continuous and differentiable. When it is a system of nonlinear equations, some numerical methods can be used in vector form [14,15]. But the formation of large matrices and the need of the differentiability of the functions are still required. Hence, researches have been carried out to seek alternative methods.

A modified Newton's method with a singular Jacobian is a study done by Jose L. Hueso et al. [16]. In order to get quadratic convergence, the Newton's method requires a nonsingular Jacobian matrix. To overcome the difficulty of the use of nonsingular Jacobian, they have presented a variant where the Jacobian can be singular. The accuracy of the approximations are high which is 10^{-12}. The variant is better in terms of complexity when compared with the classical Newton's method, but still carry the requirement of the calculations such as the need of derivatives and continuity and proper initial guesses.

Complexities in numerical approaches geared the researchers toward the problem in a different way. In a number of researches, the problem is treated as an optimization problem. Nikos E. Mastorakis has done a study on solving non-linear equations via Genetic Algorithms (GA) [17]. The objective functions tested in the paper are min $(f_1^2 + f_2^2 + \cdots + f_n^2)$ and min $(|f_1| + |f_2| + \cdots + |f_n|)$. Only a single example is used to test the method which does not illustrate the quality of the approach sufficiently. But here the advantage of not considering the differentiability is present.

Crina Grosan and Ajith Abraham use a new approach to solve systems of nonlinear equations [5]. The study has focused on treating the system of nonlinear equations as a multi-objective optimization problem where every equation represents an objective function whose goal is to minimize the difference between the right and the left terms of the corresponding equation in the system. Since it is multi-objective, to compare the solutions, the concept of Pareto dominance relationship has been used. GA is used as an evolutionary algorithm to solve the problem. The study has clearly defined the used parameters and techniques. They have tested systems up to 20 variables. Their approach was capable of obtaining several solutions within a single run but the accuracy of a solution is around 10^{-1}. The results have revealed that the proposed approach is able to deal with high dimensional nonlinear systems. Since the problem is handled as a multi-objective one, with the increase of the number of variables the complexity of the problem becomes high which in our case is not a problem.

Harmony search is another new optimization algorithm which has been used to solve systems of nonlinear equations. The harmony search combined with differential evolution, termed as a differential best harmony search was proposed in the study [10]. The results were shown with a set of lower dimensional problems. However to solve two variable nonlinear systems, they have used a large number of function evaluations with a population size of 200 (approximately 30,000) when compared with other approaches.

Particle swarm optimization (PSO), an oldest of optimization algorithms, is used in several studies to solve systems of nonlinear equations. Majid Jaberipour et al. in their study have used PSO in an efficient way [8]. The problem was converted into a single objective optimization problem where the objective is to minimize the sum of squares of function values. To overcome the tendency of being trapped in local minima and slow convergence, the traditional PSO has been modified by updating each particle. Several benchmark problems were used to illustrate the idea. Comparisons with other algorithms conclude that the new modification works well to find solutions of a system yet no attention was paid to find multiple solutions of a system.

Ariyaratne et al. have used firefly algorithm(FA) to solve single variable nonlinear equations [18]. The original FA was modified with an archive and a flag to determine all possible root approximations within a reasonable interval. They have also used the self-tuning property introduced for the optimization algorithms to optimize the parameters of the FA while finding root approximations [19]. The accuracy of a root is set to be 10^{-3}. The results revealed that the modified self-tuning firefly algorithm works well in solving single variable nonlinear equations even for the complex roots.

Use of Multi population Parallel Competitive Algorithm (PICA) to solve systems of nonlinear equations is a recent study which also goes under evolutionary approaches [20]. They have presented the method as a solution to the weaknesses in other evolutionary approaches such as large population sizes and slow convergence. The method seems to be well formed acquiring higher accuracies than other methods. However they haven't paid attention on simultaneous root approximations. The new method was tested with systems having maximum of eight equations and eight unknowns. Less attention was paid to explain the proposed method and its parameters which make the reproduction of the algorithm a difficult task.

There are a few studies focusing on finding multiple root approximations for systems of nonlinear equations. Locating Multiple Optimal Solutions of Nonlinear Equation Systems Based on Multi-objective Optimization is a study where problem is treated as a multi-objective optimization problem [21]. But here the conversion is different where multi-objective is achieved as a bi-objective situation which can divide into two parts. This transformation is combined with GA to provide the solution. The primary focus of this study is the transformation technique, and therefore less information are given on obtained results. However the indicators they have used show that the method is not so successful in providing all root approximations of a particular system within a single run.

Another study has used a variant of differential evolution to implement the same weighted bi-objective transformation technique for the same purpose of root approximation [22]. The solution seems to be working well but here also the number of function evaluations even for a system of two variables is very high (50,000) (in MODFA for a system of two variables maximum function evaluations were around 6000 (No of fireflies 30 and No of iterations 200)). The accuracy of the obtained roots are also not mentioned. The average percentage of the optimal solutions found over all the runs (PR) and the percentage of the successful runs (SR) are not 100% for many instances giving evidence that the problem does not perform well for all those instances. However the attempt to find multiple solutions within a single run is admirable. This approach has been modified in a recent research work by combining the strengths of the repulsion technique, diversity preservation mechanism, and adaptive parameter control of the differential evolution (RADE) [23]. They have used repulsion technique to find roots repulsing the regions surrounding the previously found roots. To explore the search space to find new roots, the diversity preservation mechanism has been adopted. Adaptive parameter control is used to enhance the performance of the differential evolution. Although the algorithm seems to be complex with combination of different techniques, they have successfully implemented it for many nonlinear systems including 20 variable problems. Function evaluations are comparatively higher in number which is in between 50,000 and 500,000 even for 3 variable nonlinear systems (For MODFA it was around 10,000 (No of fireflies 50 and No of iterations 200))). Compared to that, MODFA is simple in the implementation and does not require that much of function evaluations to find solutions.

Use of repulsion technique to solve systems of nonlinear equations can be seen in another research recently, where they propose a dynamic repulsion technique, to change the repulsive radius dynamically during the run [24]. The repulsive radius proposes promising search spaces to find new roots. Combined with evolutionary algorithms, these dynamic repulsive radius' allow the method to find multiple root approximations within a single run. Being a general framework, the proposed work allow to use any evolutionary algorithm to solve systems of nonlinear equations, which is an advantage of the method. In the study, they have tested the algorithm with five EAs and three repulsion techniques. Results with 42 nonlinear systems proved the effectiveness of the proposed dynamic repulsion method.

Apart from these, Artificial Bees Colony algorithm, Cukoo Search algorithm, variants of GA and variants of Differential Evolution (DE) have been proposed in several other research papers as well [5,25–27]. Solving systems for multiple roots has also been addressed [23,28]. Heuristics were also proposed in some studies [13]. The following areas have still room for improvement.

- Solving systems in higher dimensions.
- Solving systems for multiple roots.
- Solving systems without concerning initial guesses, differentiability and continuity.
- Obtaining solutions with higher accuracy.
- Obtaining multiple approximations simultaneously.

The present work is based on a popular meta-heuristic algorithm with few modifications. The study focuses on obtaining multiple roots of a nonlinear system without concerning initial guesses, differentiability and continuity. We have addressed problems with higher dimensions, but the accuracy has not been set to be at a higher level, since the objectives are to find multiple roots, simultaneously within a single run.

3. Preliminaries

3.1. Nonlinear functions of several variables

Let $D \subset \mathbb{R}^n$ and suppose $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is a real-valued function which assigns a unique real number denoted by $f(x_1, x_2, x_3, \dots, x_n)$ to each $(x_1, x_2, x_3, \dots, x_n) \in D$. The set D is the domain of f and its range is the set of values that f takes on. That is,

$$\{f(x_1, x_2, x_3, \dots, x_n) \in \mathbb{R} \mid (x_1, x_2, x_3, \dots, x_n) \in D\}.$$

We write $z = f(x_1, x_2, x_3, \dots, x_n)$ to make explicit the value taken by f at a general point $(x_1, x_2, x_3, \dots, x_n)$. The variables $(x_1, x_2, x_3, \dots, x_n)$ are independent and z is dependent [29].

3.2. A system of nonlinear equations

A system of nonlinear equations has the form of

$$f_1(x_1, x_2, \dots, x_n) = 0$$

$$f_2(x_1, x_2, \dots, x_n) = 0$$

$$\dots$$

$$\dots$$

$$f_n(x_1, x_2, \dots, x_n) = 0$$

$$\underline{F}(\underline{X}) = \underline{0}$$

where each f_i is a nonlinear function of n variables. Some of the equations can be linear, but not all of them. Finding a solution for a nonlinear system of equations $\underline{F}(\underline{X})$ involves finding a solution such that every equation in the nonlinear system is 0, i.e. this system of n - nonlinear equations in n unknowns can alternatively be written as $\underline{F}(\underline{X}) = \underline{0}$ by defining a vector valued function $\underline{F}(\underline{X})$.

The solutions we are interested in are those points (if any) that are simultaneously zeros of $f_i, i = 1, \dots, n$. A system of nonlinear equations can have multiple solutions.

Consider the following system of two variables.

$$f_1(x_1, x_2) : x_1^4 + x_2^4 - 67 = 0$$
$$f_2(x_1, x_2) : x_1^3 - 3x_1x_2^2 + 35 = 0 \quad (3.1)$$

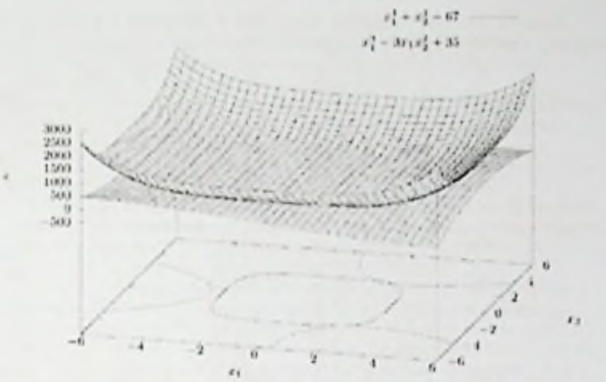

Fig. 1. Surface plot and the contour plot for the nonlinear system 3.1 at $z = 0$.

The cross sectional view (contour at $z = 0$) and the surface plot of the two functions of Eq. (3.1) are shown in Fig. 1. For two dimensional nonlinear systems, contour plot at $z = 0$ gives a better view to get an idea about the location of roots within the specified state space.

This system has 2 roots as [1.9358, −2.6976] and [1.9416, 2.6954] (approximations), within the specified region.

Here, we focus on the nonlinear systems having equal number of variables and equations which are known as square systems. Underdetermined (more variables than equations) and Overdetermined (more equations than variables) systems were not used to test the proposed MODFA.

3.3. Firefly algorithm (FA)

Firefly is a winged beetle commonly known as the lightning bug due to the charming light it emits. The light is used to attract mates or preys. Biological studies reveal many interesting factors about fireflies' life style [30]. Focusing on their flashing behavior, the FA was developed by Xin-She-Yang in 2009 [12]. The algorithm basically assumes the following.

- Fireflies' attraction to each other is gender independent.
- Attractiveness is proportional to the brightness, for any two fireflies, the less bright one is attracted and moves toward the brighter one; the brightness decrease as the distance increases; If there is no brighter one than a particular firefly, it moves randomly.
- The brightness of a firefly is determined by the value of the problem specific objective function.

As many meta-heuristics, the initial population for the particular problem is generated randomly. In FA also, the parameter set should be specified properly. After these initial steps, the fireflies in the population start moving towards brighter fireflies according to the following equation.

$$x_i = x_i + \beta(x_j - x_i) + \alpha(rand - 0.5) \quad (3.2)$$

where

$$\beta = \beta_0 . e^{-\gamma r^2} \quad (3.3)$$

Here x_i and x_j refer to two fireflies. β is the attraction between two fireflies and α is the parameter controlling the step size. β_0 is the attraction at $r = 0$, where r is the distance between two fireflies. The three terms in Eq. (3.2) represent the contribution from the current firefly, attraction between two fireflies and a randomization term respectively. The equation supports both exploitation and exploration. α plays an important role in the randomization process, which is from Uniform or Gaussian distribution. To control the randomness, after each iteration, Yang

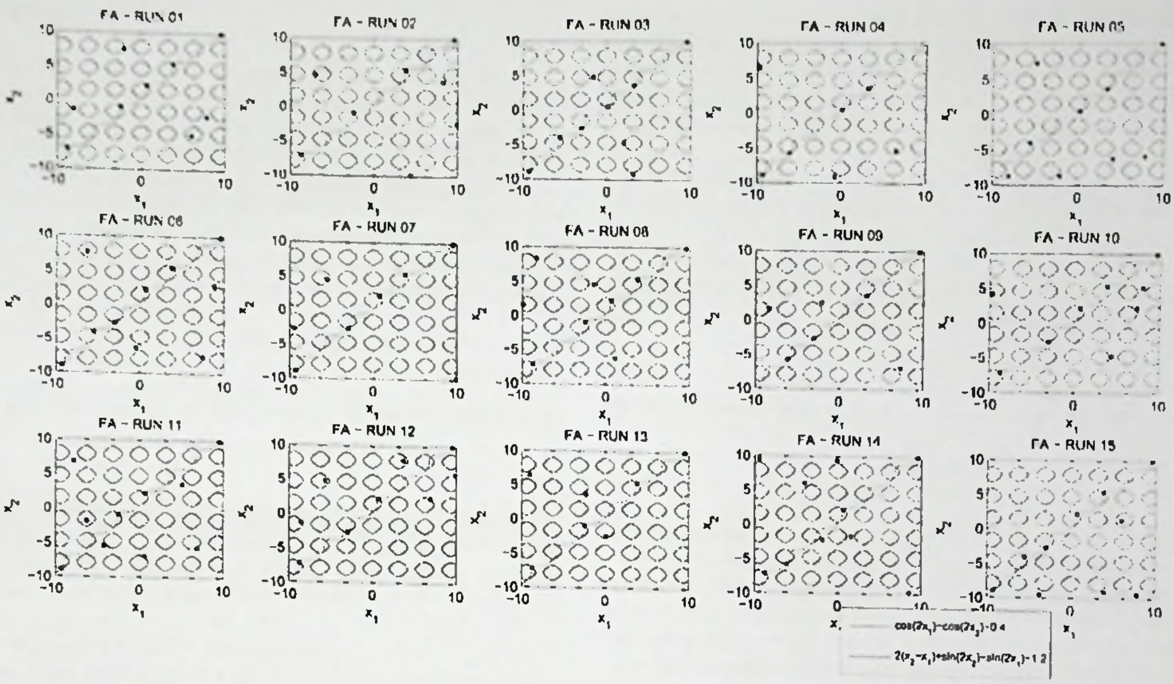

Fig. 2. Convergence of roots of the nonlinear system F08 over 15 runs of FA.

has used δ, the randomness reduction factor which reduces α according to Eq. (3.4).

$$\alpha = \alpha.\delta \quad where \quad \delta \in [0,1] \tag{3.4}$$

FA, as a well established performer in the world of meta-heuristics marked its remarkable capability of handling optimization problems. The applications of FA and its variants are diverse and up to now thousands of researches have been carried out by using FA, covering many different areas. For example, FA has been applied in different areas such as clustering [31], image processing [32,33], forecasting [34,35] and many more. Different variants for the FA have also been proposed in discretizing FA appropriately [36,37], hybrid FA with other techniques [38–40], multi-objective FA algorithms [41–43] and so on. Complete reviews on firefly algorithm and its applications can be found in Refs. [44–46]. Yang et al., in 2013 introduced a framework for self-tuning algorithms and it was originally implemented with the firefly algorithm successfully [19].

4. Modified firefly algorithm to solve systems of nonlinear equations of several variables (MODFA)

The original firefly algorithm is slightly modified to suit the situations with multiple optimum solutions, since we are trying to find more than one root approximation of a given system of nonlinear equations simultaneously.

For the purpose of solving nonlinear systems, we have tried several other nature inspired algorithms as well. Since our objective is different from improving the accuracy of the roots, by experiments, FA is selected as the best available performer. As many optimization algorithms, firefly algorithm also improves solutions by both exploitation and exploration (see Eq. (3.2)), where for the problem addressed in this paper needs more exploration. The original FA has more exploration power compared with other natural algorithms so that when solving a particular nonlinear system using the original FA, it may provide several root approximations within a single run, but is limited. Within several runs it may provide several more roots but it depends on the problem and the number of roots. There is a probability that one may not get many root approximations within several runs as well. Therefore, as improvements, several modifications were included to advance the exploration property of the algorithm. These modifications include an archive to collect root approximations and a flag to determine poor iterations to improve the exploration property. When exploration is improved the algorithm is able to explore a vast search space allowing it to find many root approximations. To illustrate the performance of the modification over the original FA, we have solved nonlinear systems using FA and MODFA. FA was executed for several runs and for the MODFA results were taken within a single run (500 iterations). Figs. 2 and 3 show the obtained results for the nonlinear system F08 from two algorithms.

As shown in Fig. 2, FA within 15 runs was able to give maximum of 6 roots approximations at a time. But over the 15 runs there are some roots that have not been found in any run. If we run further we may find those root approximations as well, but it may be by chance and there is no guarantee. But MODFA was capable of giving all 13 root approximations within a single run in the same circumstances (see Fig. 3). So that it can be stated that the modification appears to be well suited for the problem addressed here.

In our method, each firefly in the population represents a possible approximation to the roots of the nonlinear system within the predefined state space.

`Firefly[i] = [root approximations]`.

For an example, consider the `Nonlinear system (3.1)`. A possible initial firefly would be.

`Firefly[i] = [0.7942, -2.4148]` within $[-3, 3] \times [-3,3] \subset \mathbb{R}^2$. If the algorithm uses 10 fireflies, then each firefly carries a root approximation to the system at the beginning. The MODFA will then work as follows.

- Each firefly's fitness is calculated using the objective function. Here the fitness is calculated considering the objective of the problem, which is finding roots. The objective function for the system is defined with the help of vector norm concept. For a particular firefly, we evaluate the functions in the system using its root approximations. For an example if we evaluate the `Nonlinear System(3.1)` at `[0.7942, -2.4148]` we get f_1 and

[15] W.B. Gragg, G.W. Stewart, A stable variant of the secant method for solving nonlinear equations, SIAM J. Numer. Anal. 13 (3.6) (1976) 889–903. https://doi.org/10.1137/0713070.

[16] J.L. Hueso, E. Martínez, J.R. Torregrosa, Modified Newton's method for systems of nonlinear equations with singular jacobian, J. Comput. Appl. Math. 224 (3.1) (2009) 77–83. https://doi.org/10.1016/j.cam.2008.04.013, http://www.sciencedirect.com/science/article/pii/S0377042708001787.

[17] N.E. Mastorakis, Solving non-linear equations via genetic algorithms, in: Proceedings of the 6th WSEAS International Conference on Evolutionary Computing, EC'05, World Scientific and Engineering Academy and Society (WSEAS), Stevens Point, Wisconsin, USA, 2005, pp. 24–28. http://dl.acm.org/citation.cfm?id=1904128.1904133.

[18] M.K.A. Ariyaratne, T.G.I. Fernando, S. Weerakoon, A self-tuning modified firefly algorithm to solve univariate nonlinear equations with complex roots, in: 2016 IEEE Congress on Evolutionary Computation (CEC), 2016, pp. 1477–1484, https://doi.org/10.1109/CEC.2016.7743964.

[19] X.-S. Yang, S. Deb, M. Loomes, M. Karamanoglu, A framework for self-tuning optimization algorithm, Neural Comput. Appl. 23 (7–8) (2013) 2051–2057.

[20] A. Majd, M. Abdollahi, G. Sahebi, D. Abdollahi, M. Daneshtalab, J. Plosila, H. Tenhunen, Multi-population parallel imperialist competitive algorithm for solving systems of nonlinear equations, in: 2016 International Conference on High Performance Computing Simulation (HPCS), 2016, pp. 767–775. https://doi.org/10.1109/HPCSim.2016.7568412.

[21] W. Song, Y. Wang, H.X. Li, Z. Cai, Locating multiple optimal solutions of nonlinear equation systems based on multiobjective optimization, IEEE Trans. Evol. Comput. 19 (3.3) (2015) 414–431, https://doi.org/10.1109/TEVC.2014.2336865.

[22] W. Gong, Y. Wang, Z. Cai, S. Yang, A weighted biobjective transformation technique for locating multiple optimal solutions of nonlinear equation systems, IEEE Trans. Evol. Comput. 21 (3.5) (2017) 697–713, https://doi.org/10.1109/TEVC.2017.2670779.

[23] W. Gong, Y. Wang, Z. Cai, L. Wang, Finding multiple roots of nonlinear equation systems via a repulsion-based adaptive differential evolution, IEEE Trans. Syst., Man, Cybern.: Syst. (2018) 1–15, https://doi.org/10.1109/TSMC.2018.2828018.

[24] Z. Liao, W. Gong, X. Yan, L. Wang, C. Hu, Solving nonlinear equations system with dynamic repulsion-based evolutionary algorithms, IEEE Trans. Syst., Man, Cybern.: Syst. (2018) 1–12. https://doi.org/10.1109/TSMC.2018.2852798.

[25] Y.G. Li, R.H. Zhou, An improve cuckoo search algorithm for solving nonlinear equation group, in: Material Science, Civil Engineering and Architecture Science, Mechanical Engineering and Manufacturing Technology II, Trans Tech Publications, 2014, pp. 2121–2124, https://doi.org/10.4028/www.scientific.net/AMM.651-653.2121. Vol. 651 of Applied Mechanics and Materials.

[26] H. Ren, L. Wu, W. Bi, I.K. Argyros, Solving nonlinear equations system via an efficient genetic algorithm with symmetric and harmonious individuals, Appl. Math. Comput. 219 (23) (2013) 10967–10973, https://doi.org/10.1016/j.amc.2013.04.041.

[27] A. Rovira, M. Valdés, J. Casanova, A new methodology to solve non-linear equation systems using genetic algorithms. application to combined cyclegas turbine simulation, Int. J. Numer. Methods Eng. 63 (10) (2005) 1424–1435, https://doi.org/10.1002/nme.1267, https://doi.org/10.1002/nme.1267.

[28] C. Grosan, A. Abraham, A. Gelbukh, Evolutionary Method for Nonlinear Systems of Equations, Springer Berlin Heidelberg, Berlin, Heidelberg, 2006, pp. 283–293, https://doi.org/10.1007/11925231_27, https://doi.org/10.1007/11925231_27.

[29] H. Nishani, Improved Newton's Method to Solve Systems of Nonlinear Equations, Master's thesis, Faculty of Graduate Studies, University of Sri Jayewardenepura, under the supervision of S. Weerakoon, T.G.I. Fernando and M. Liyanage, April 2015.

[30] A. South, K. Stanger-Hall, M.-L. Jeng, S.M. Lewis, Correlated evolution of female neoteny and flightlessness with male spermatophore production in fireflies (coleoptera: lampyridae), Evolution 65 (3.4) (2011) 1099–1113, https://doi.org/10.1111/j.1558-5646.2010.01199.x URL http://dx.doi.org/10.1111/j.1558-5646.2010.01199.x.

[31] J. Senthilnath, S. Omkar, V. Mani, Clustering using firefly algorithm: performance study, Swarm Evol. Comput. 1 (3.3) (2011) 164–171. https://doi.org/10.1016/j.swevo.2011.06.003, http://www.sciencedirect.com/science/article/pii/S2210650211000265.

[32] T. Kanimozhi, K. Latha, An integrated approach to region based image retrieval using firefly algorithm and support vector machine, Neurocomputing 151 (2015) 1099–1111. https://doi.org/10.1016/j.neucom.2014.07.078, http://www.sciencedirect.com/science/article/pii/S0925231214013332.

[33] V. Rajinikanth, M. Couceiro, Rgb histogram based color image segmentation using firefly algorithm, Procedia Comput. Sci. 46 (2015) 1449–1457 proceedings of the International Conference on Information and Communication Technologies, ICICT 2014, 3-5 December 2014 at Bolgatty Palace & Island Resort, Kochi, India, https://doi.org/10.1016/j.procs.2015.02.064, http://www.sciencedirect.com/science/article/pii/S1877050915001283.

[34] L. Xiao, W. Shao, T. Liang, C. Wang, A combined model based on multiple seasonal patterns and modified firefly algorithm for electrical load forecasting, Appl. Energy 167 (2016) 135–153. https://doi.org/10.1016/j.apenergy.2016.01.050, http://www.sciencedirect.com/science/article/pii/S0306261916300307.

[35] I.A. Ibrahim, T. Khatib, A novel hybrid model for hourly global solar radiation prediction using random forests technique and firefly algorithm, Energy Convers. Manag. 138 (2017) 413–425. https://doi.org/10.1016/j.enconman.2017.02.006, http://www.sciencedirect.com/science/article/pii/S0196890417301036.

[36] E. Osaba, X. Yang, F. Diaz, E. Onieva, A.D. Masegosa, A. Perallos, A Discrete Firefly Algorithm to Solve a Rich Vehicle Routing Problem Modelling a Newspaper Distribution System with Recycling Policy, 2017. CoRR abs/1604.04146. arXiv:1604.04146 http://arxiv.org/abs/1604.04146.

[37] G.K. Jati, Suyanto, Evolutionary discrete firefly algorithm for travelling salesman problem, in: A. Bouchachia (Ed.), Adaptive and Intelligent Systems, Springer Berlin Heidelberg, Berlin, Heidelberg, 2011, pp. 393–403.

[38] L. Zhang, L. Liu, X.-S. Yang, Y. Dai, A Novel Hybrid Firefly Algorithm for Global Optimization, 2016, p. 11.

[39] X. Xia, L. Gui, G. He, C. Xie, B. Wei, Y. Xing, R. Wu, Y. Tang, A hybrid optimizer based on firefly algorithm and particle swarm optimization algorithm, J. Comput. Sci. 26 (2018) 488–500. https://doi.org/10.1016/j.jocs.2017.07.009, http://www.sciencedirect.com/science/article/pii/S1877750317303848.

[40] R.K. Sahu, S. Panda, S. Padhan, A hybrid firefly algorithm and pattern search technique for automatic generation control of multi area power systems, Int. J. Electr. Power Energy Syst. 64 (2015) 9–23. https://doi.org/10.1016/j.ijepes.2014.07.013, http://www.sciencedirect.com/science/article/pii/S0142061514004451.

[41] X.-S. Yang, Multiobjective firefly algorithm for continuous optimization, Eng. Comput. 29 (3.2) (2013) 175–184, https://doi.org/10.1007/s00366-012-0254-1, https://doi.org/10.1007/s00366-012-0254-1.

[42] C. Zhao, C. Wu, J. Chai, X. Wang, X. Yang, J.-M. Lee, M.J. Kim, Decomposition-based multi-objective firefly algorithm for rfid network planning with uncertainty, Appl. Soft Comput. 55 (2017) 549–564. https://doi.org/10.1016/j.asoc.2017.02.009, http://www.sciencedirect.com/science/article/pii/S1568494617300789.

[43] L. Lv, J. Zhao, J. Wang, T. Fan, Multi-objective firefly algorithm based on compensation factor and elite learning, Future Gener. Comput. Syst. (2019). https://doi.org/10.1016/j.future.2018.07.047, http://www.sciencedirect.com/science/article/pii/S0167739X18313955.

[44] I. Fister, I. Fister Jr., X.-S. Yang, J. Brest, A comprehensive review of firefly algorithms, Swarm Evol. Comput. 13 (2013) 34–46.

[45] S.L. Tilahun, J.M.T. Ngnotchouye, Firefly algorithm for discrete optimization problems: a survey, KSCE J. Civil Eng. 21 (3.2) (2017) 535–545, https://doi.org/10.1007/s12205-017-1501-1, https://doi.org/10.1007/s12205-017-1501-1.

[46] S.L. Tilahun, J.M.T. Ngnotchouye, N.N. Hamadneh, Continuous versions of firefly algorithm: a review, Artif. Intell. Rev. (2017), https://doi.org/10.1007/s10462-017-9568-0 https://doi.org/10.1007/s10462-017-9568-0.

[47] X.-S. Yang, X.-S. He, Why the Firefly Algorithm Works? Springer International Publishing, Cham, 2018, pp. 245–259, https://doi.org/10.1007/978-3-319-67669-2_11, https://doi.org/10.1007/978-3-319-67669-2_11.

[48] I. Tsoulos, A. Stavrakoudis, On locating all roots of systems of nonlinear equations inside bounded domain using global optimization methods, Nonlinear Anal. R. World Appl. 11 (3.4) (2010) 2465–2471. https://doi.org/10.1016/j.nonrwa.2009.08.003, http://www.sciencedirect.com/science/article/pii/S1468121809002508.

[49] S. Effati, A. Nazemi, A new method for solving a system of the nonlinear equations, Appl. Math. Comput. 168 (3.2) (2005) 877–894. https://doi.org/10.1016/j.amc.2004.09.029, http://www.sciencedirect.com/science/article/pii/S0096300304006460.

[50] H. Oliveira, A. Petraglia, Solving nonlinear systems of functional equations with fuzzy adaptive simulated annealing, Appl. Soft Comput. 13 (11) (2013) 4349–4357. https://doi.org/10.1016/j.asoc.2013.06.018, http://www.sciencedirect.com/science/article/pii/S156849461300207X.

[51] I.M. El-Emary, M.A. El-Kareem, Towards using genetic algorithm for solving nonlinear equation systems, World Appl. Sci. J. 5 (3.3) (2008) 282–289.

[52] A. Ouyang, Y. Zhou, Q. Luo, Hybrid particle swarm optimization algorithm for solving systems of nonlinear equations, in: 2009 IEEE International Conference on Granular Computing, 2009, pp. 460–465, https://doi.org/10.1109/GRC.2009.5255079.

[53] C. Grosan, A. Abraham, Multiple solutions for a system of nonlinear equations, Int. J. Innov. Comput., Inf. Contr. 91 (2008) 37–47, https://pdfs.semanticscholar.org/6115/748d97024691f9f712a779a1a1a7308185e1.pdf.

[54] H. Hong, V. Stahl, Safe starting regions by fixed points and tightening, Computing 53 (3.3) (1994) 323–335, https://doi.org/10.1007/BF02307383, https://doi.org/10.1007/BF02307383.

[55] R.E. Moore, Methods and Applications of Interval Analysis, SIAM, 1979.

[56] A. Morgan, Solving Polynomial Systems Using Continuation for Engineering and Scientific Problems, Society for Industrial and Applied Mathematics, 2009https://doi.org/10.1137/1.9780898719031, http://epubs.siam.org/doi/pdf/10.1137/1.9780898719031, http://epubs.siam.org/doi/abs/10.1137/1.9780898719031.

[57] C.G. Broyden, A class of methods for solving nonlinear simultaneous equations, Math. Comput. 19 (1965) 577–593. http://www.ams.org/journals/mcom/1965-19-092/S0025-5718-1965-0198670-6/home.html.

[58] S. Das, S.S. Mullick, P. Suganthan, Recent Advances in Differential Evolution - an Updated Survey, Swarm and Evolutionary Computation, vol. 27, 2016, pp. 1–30. https://doi.org/10.1016/j.swevo.2016.01.004, http://www.sciencedirect.com/science/article/pii/S2210650216000146.

[59] C.-W. Chiang, W.-P. Lee, J.-S. Heh, A 2-opt based differential evolution for global optimization, Appl. Soft Comput. 10 (3.4) (2010) 1200–1207 optimisation Methods & Applications in Decision-Making Processes, https://doi.org/10.1016/j.asoc.2010.05.012, http://www.sciencedirect.com/science/article/pii/S1568494610001134.

[60] W. Daniel, Applied Nonparametric Statistics, the Duxbury Advanced Series in Statistics and Decision Sciences, PWS-Kent Publ., 1990, https://books.google.lk/books?id=0hPvAAAAMAAJ.

[61] I. Minitab, Minitab 17 Statistical Software [Computer Software], 2010, www.minitab.com.

[62] M. Abdollahi, A. Isazadeh, D. Abdollahi, Imperialist competitive algorithm for solving systems of nonlinear equations, Comput. Math. Appl. 65 (12) (2013) 1894–1908, https://doi.org/10.1016/j.camwa.2013.04.010, http://www.sciencedirect.com/science/article/pii/S0898122113002241.

[63] M. Abdollahi, A. Bouyer, D. Abdollahi, Improved cuckoo optimization algorithm for solving systems of nonlinear equations, J. Supercomput. 72 (3.3) (2016) 1246–1269, https://doi.org/10.1007/s11227-016-1660-8, https://doi.org/10.1007/s11227-016-1660-8.

[64] [illegible], S. Weerakoon, T. Fernando, M. Liyanage, Weerakoon-Fernando method with accelerated third-order convergence for systems of nonlinear equations, Int. J. Math. Model. Numer. Optim. 8 (3) (2018) 282–304.

[65] S. Weerakoon, T. Fernando, A variant of Newton's method with accelerated third-order convergence, Appl. Math. Lett. 13 (3.8) (2000) 87–93, https://doi.org/10.1016/S0893-9659(00)00100-2, http://www.sciencedirect.com/science/article/pii/S0893965900001002.

[66] H. Ramos, M. Monteiro, A new approach based on the Newton's method to solve systems of nonlinear equations, J. Comput. Appl. Math. 318 (2017) 3–13 computational and Mathematical Methods in Science and Engineering CMMSE-2015, https://doi.org/10.1016/j.cam.2016.12.019, http://www.sciencedirect.com/science/article/pii/S037704271630620X.

[67] X. Xiao, H. Yin, A simple and efficient method with high order convergence for solving systems of nonlinear equations, Comput. Math. Appl. 69 (10) (2015) 1220–1231, https://doi.org/10.1016/j.camwa.2015.03.018, http://www.sciencedirect.com/science/article/pii/[illegible].