JOURNAL OF UNIVERSAL MATHEMATICS
VOL.8 NO.1 PP.52-69 (2025)
ISSN-2618-5660
DOI: 10.33773/jum.1551816

PERFORMANCE COMPARISON OF FIXED-POINT ITERATION METHOD AND TEACHING-LEARNING BASED OPTIMISATION: A STUDY ON NONLINEAR EQUATION SYSTEMS

BAHAR DEMİRTÜRK, BAYRAM KÖSE and ŞÜKRAN KONCA

0000-0002-5911-5190, 0000-0003-0256-5921 and S0000-0003-4019-958X

ABSTRACT. This study focuses on two main objectives. Firstly, the similarities and differences between the mathematically based fixed point iteration method and the metaheuristic teaching-learning based optimisation method are presented. Secondly, the performance of these two methods in finding solutions of a complex system of linear equations is compared. In this way, other researchers will be able to make a comparison between the results previously discussed by the authors in [2] and [3], respectively, and have an idea about choosing the required optimisation method using these results in their future research.

1. INTRODUCTION

Root-finding problems are one of the most frequently encountered and critically important topics in mathematics and engineering. Finding solutions to nonlinear equations plays an important role in both theoretical studies and practical applications [4-6]. However, since analytical solutions are not possible in many cases, iterative methods come into play. These methods use an iterative process to find the roots of complex equations and are evaluated by performance criteria such as convergence rates and accuracy levels.

Traditional optimization methods usually involve mathematical modelling, using knowledge of derivatives as well as various techniques such as linear programming, integer programming, genetic algorithms. These methods seek solutions to optimize a given objective function under a set of constraints. However, these methods may not be sufficient for some problems. For example, in complex dynamic systems, the problem structure and constraints may change over time or be uncertain. Also, traditional optimization methods may be limited in terms of computational power and data processing capabilities when dealing with large datasets. Different optimization methods have been developed to overcome the limitations of traditional approaches and produce more efficient solutions [7-10]. These methods include data collection, analysis and learning processes. One of these methods is the Teaching-Learning Based Optimization (TLBO) algorithm, which uses the information obtained from past data in the teaching process to support future decisions [11].

In this paper, we investigate the performance of two different iterative methods - the Fixed-Point Iterative Method and the Teaching-Learning Optimization Algorithm (TLBO) - on Capra and Canale's (2002) system of nonlinear equations given in [1]. The Fixed-Point Iterative Method is a classical and widely used technique, based on a simple iterative process to find the root of the equation. On the other

Date: **Received**: 2024-09-17; **Accepted**: 2024-12-02.
Key words and phrases. Root finding, Fixed point iteration method, Teaching-learning algorithm, Optimization techniques.

hand, the TLBO algorithm is a more modern optimization method inspired by the teaching and learning processes in nature.

The aim of this study is to compare the performance of these two methods, to reveal their common and different aspects and to determine which method is more effective in which situations. For this purpose, various tests were performed on Capra and Canale's system of nonlinear equations and the results obtained were analysed graphically. This study provides important findings for understanding the performance of different iterative methods on nonlinear equations and sheds light on future research.

1.1 Fixed-Point Iteration Method

The fixed-point iteration method was first used by the German mathematician L.E.J. Brouwer in the early 1900s and is used in many areas of mathematics, especially in numerical analysis. This method is used to find approximate solutions of linear equations as well as approximate solutions of nonlinear systems of equations.

In this method, which is used to solve an equation of the form $f(x)=0$, let the given equation be expressed by the function $x=g(x)$. Let the point x_0 be the first estimated point and the point $x=x_0$ be chosen such that $|g'(x)|<1$. By this we mean that convergence is absolute, i.e. it always converges towards the root. In this case, with successive iteration

$$x_1 = g(x_0)$$
$$x_2 = g(x_1)$$
$$\vdots$$
$$x_n = g(x_{n-1})$$

iterative method is obtained.

The absolute difference between the root found and the previous root gives the absolute error, $E_0 = |x_1 - x_0|$, $E_1 = |x_2 - x_1|, \ldots, E_n = |x_{n+1} - x_n|$ be defined as the zeroth, first and nth absolute errors respectively. In this case, one can see the following

$$\frac{E_1}{E_0} = \frac{|x_2 - x_1|}{|x_1 - x_0|} = \frac{|g(x_1) - g(x_0)|}{|x_1 - x_0|}$$
$$\frac{E_2}{E_1} = \frac{|x_3 - x_2|}{|x_2 - x_1|} = \frac{|g(x_2) - g(x_1)|}{|x_2 - x_1|}$$
$$\vdots$$
$$\frac{E_{n+1}}{E_n} = \frac{|x_{n+2} - x_{n+1}|}{|x_{n+1} - x_n|} = \frac{|g(x_{n+1}) - g(x_n)|}{|x_{n+1} - x_n|}$$

Therefore

$$\lim_{x_1 \to x_0} \frac{E_1}{E_0} = \lim_{x_1 \to x_0} \left| \frac{g(x_1) - g(x_0)}{x_1 - x_0} \right| = |g'(x_0)|$$
$$\lim_{x_2 \to x_1} \frac{E_2}{E_1} = \lim_{x_2 \to x_1} \left| \frac{g(x_2) - g(x_1)}{x_2 - x_1} \right| = |g'(x_1)|$$
$$\vdots$$
$$\lim_{x_{n+1} \to x_n} \frac{E_{n+1}}{E_n} = \lim_{x_{n+1} \to x_n} \left| \frac{g(x_{n+1}) - g(x_n)}{x_{n+1} - x_n} \right| = |g'(x_n)|$$

can be written. It can be seen that for a given iteration number n, if $|g'(x_n)| < 1$ while $n \to \infty$, then x_n converges to real root. In particular, the fixed-point iteration method also gives an idea that if $|g'(x_0)| < 1$ for $x_1 \to x_0$ then the initial solution can be used to reach the conclusion.

The main idea behind the choice of fixed-point iteration functions is to decompose the equation $f(x) = 0$ appropriately and replace it with two equations of the form $y_1 = g(x)$ and $y_2 = h(x)$. The generated system is solved sequentially. Here, the following equation can be written for $g(x)$ and $h(x)$, which are parts of the equation:

$$f(x) = g(x) - h(x) = 0.$$

By doing this, the number of equations to be solved is doubled, but the equations are simplified. One of them can even be directly equal to x or solved with respect to x. In the application of the method, iteration starts with an initial value that is assumed to be close to the root. The first equation is either equal to x_0 or x_1 is found by substituting x_0. In the second equation, x_2 is calculated using x_1 and this process is continued until the desired approximate root value is reached. For this, the following algorithm is applied.

Step 1. An initial value x_0 close to the root is estimated.
Step 2. The equation $f(x) = 0$ is rearranged in the form of $x = g(x)$.
Step 3. A new value for the root is calculated in the equation $x_{i+1} = g(x_i)$.
Step 4. If $\varepsilon_a = \left| \frac{x_{i+1} - x_i}{x_{i+1}} \right| .100 \leq \varepsilon_s$ then stop, otherwise go to Step 3 by taking $x_i = x_{i+1}$.

In the fixed-point method, there is always the possibility of divergence as well as convergence. Convergence and divergence are shown graphically in Figure 1 [13].

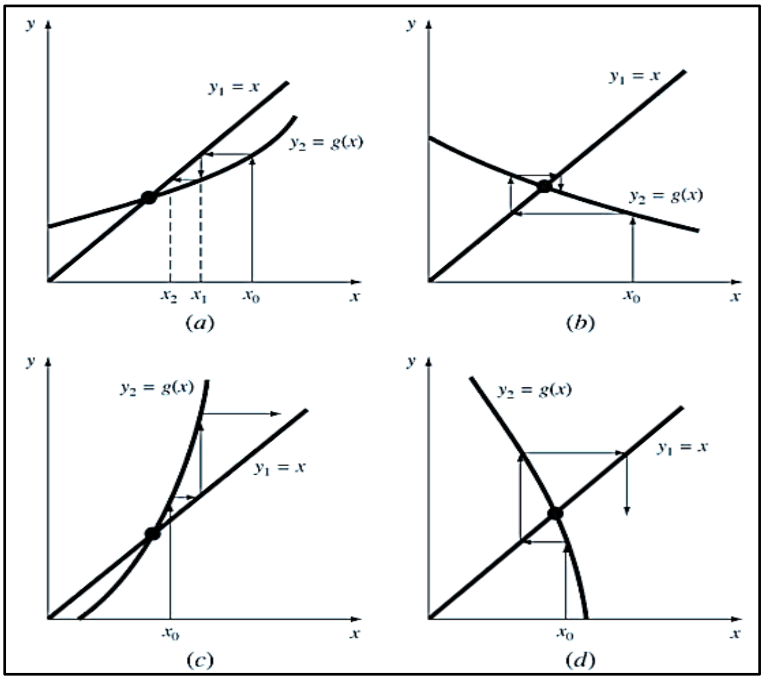

Figure 1. The fixed-point method's a) convergence case, b) convergence case c) divergence case, d) divergence case

Figure 1 (a) and Figure 1 (b) are graphical representations of the convergence of fixed-point iteration and Figure 1 (c) and Figure 1 (d) are graphical representations of the divergence of fixed-point iteration, where (a) and (b) are called monotonic graphs and (c) and (d) are called oscillating or spiral graphs. Convergence can be realized under the condition $|g'(x)| < 1$.

In Figure 1, the point x_0 is used as the initial value. In the graph (d), when moving from the point x_0 to the line $y_1 = x$ from the point where the curve $y_2 = g(x)$ is reached, it seems to be approached to the root value, but then when the iteration is continued, in other words, when trying to approach the intersection point of the curves using the newly found approximate root value, it is seen that it moves away from this point. These situations can be encountered from time to time in the constant iteration method. Similarly, in graph (b), each iteration gets closer and closer to the root and the error value decreases with each step.

Note that convergence occurs when the absolute value of the slope of the function $y_2 = g(x)$ is smaller than the slope of the function $y_1 = x$. If convergence occurs, the error at each step is the same or smaller than the error at the previous step. Therefore, fixed point iteration has linear convergence.

1.2 Teaching-Learning Based Optimization Algorithm (TLBO)

Learning and teaching based optimization (TLBO) can be defined as an approach that combines learning and teaching components related to the optimization problem. In the learning phase, TLBO analyses data relevant to problem solving and extracts knowledge and patterns by learning from this data. The learning process can converge to the optimal solution by using strategies, constraints or other factors to solve the problem with necessary updates to the teacher's experience and results. The learning process often involves statistical analysis, machine learning or artificial intelligence techniques.

A teaching-learning based optimization method is considered by Rao et al. [12]. As the solution population, the operations take place with classes and students as its members. The aim is to increase the knowledge level of the students in the class in order to obtain the optimum solution. Basically, it is realized in two phases such as teaching and learning. It is represented as a matrix representing the classes and the students in the classes. Each row in the matrix corresponds to a student. The rows represent the design proposal. The analysis starts with the random assignment of sections from a pre-prepared list of profiles [12].

Learning Phase. The student who gives the best solution in the class is considered as the teacher. Accordingly, the other students are updated according to the following relationships by utilizing the teacher's knowledge. If the updated student gives a better solution than the old one, he/she replaces the old student.

Teaching Phase. The process in this phase is very similar to the previous phase. There is interaction between the students in the class. There is a process of transferring knowledge from one student with a better solution and a higher level of knowledge to another student. If the new student finds a better solution than the current student, he/she will take his/her place.

With teaching and learning based optimization, if the teaching and learning steps are considered as the interaction between teachers and students in a classroom, first the population (class size) dimensions to be evaluated are determined. Then the objective function is determined. In line with the determined objective function, the best individual (x) in the population is assigned as a teacher. The mean of the population (class) is calculated. Interaction between teacher and student is ensured. At this stage, a teacher tries to transfer information between students and increase the average result of the class. In the next stage, students try to increase their knowledge level through interaction among themselves. Students can also gain knowledge by discussing and interacting with other students. A student standing in the center of the class can communicate with those in the next row and across. The interaction will be provided in such a way that a student will learn new information if the other student has more information about him/her.

$$x_{new} = x + r.(x_{best} - T_f .x_{arithmetic-mean}) \qquad (1)$$

In Equation (1), T_f is a constant that takes the value 1 or 2. *r* represents a random number in the closed interval [0,1]. x_{new} is the new student, *x* is the best student from the previous iteration, x_{best} is the best student and $x_{arithmetic-mean}$ is the arithmetic mean of the population. With the formula given in Equation (1), the knowledge level of the population (students) is determined after the interaction between the population individuals. The best individual is then selected as the teacher. The cycle continues until the determined learning level is achieved. Learning and teaching based optimization (TLBO) is defined as an algorithm that can model the effect of learning on students in the classroom [11].

1.3. Comparison of Common and Differences between Fixed Point Iteration Method and TLBO Algorithm

While the fixed-point iteration method and the TLBO algorithm differ in their applications and specific methodologies, they undoubtedly share the following common features, especially in the context of iterative and optimisation processes.

Fixed point iteration and TLBO algorithms aim to solve different types of problems with iterative approaches. Starting with an initial guess, FPI iteratively applies a function and converges to a fixed point satisfying the condition $f(x)=x$. In contrast, TLBO is based on improving a population of solutions. TLBO consists of two phases: teaching and learning: In the teaching phase the best solution guides the process, while in the learning phase the solutions are improved by learning from each other.

Both algorithms have different convergence goals and dependencies. While fixed point iteration method focuses on fixed point finding problems, TLBO is designed to solve direct optimisation problems. In fixed point iteration method, the initial guess affects both the convergence speed and the final solution, while in TLBO the quality of the initial population determines the performance of the algorithm and the quality of the solution obtained.

The stopping criteria also differ between the two methods. Fixed point iteration stops when the difference between consecutive iterations falls below a certain threshold. TLBO usually stops when it reaches a certain number of iterations, when convergence reaches a threshold, or when the improvement rate becomes negligible.

Fixed point iteration method focuses on a single solution point by providing a mathematical approach. TLBO is a heuristic metaheuristic that iteratively evolves a population of solutions. While fixed point iteration method is mostly used in areas such as numerical analysis, equation solving and mathematical modelling, TLBO has a wide range of applications in engineering, economics and scientific optimisation problems.

As a result, fixed point iteration method has a simpler and mathematical structure, while TLBO is a complex and powerful optimisation technique inspired by the teaching-learning process. The nature, objectives and application areas of the two determine their suitability for different types of problems.

1.4. Optimization Approach for Finding the Roots

When the optimization process is used to find the roots of algebraic equations, the problem of finding the unknown values in each equation becomes an optimization problem to be solved by numerical methods. Optimization is the process of obtaining the best value of an objective function according to specified criteria. Since the numerical approach for finding roots in algebraic equations usually involves an iterative process, similarly, in finding roots with an optimization algorithm, starting from a given starting point, candidate root values are iteratively updated and reach a minimum or maximum value when the objective function is sufficiently close or a certain tolerance value is reached.

In this section, Theorem 1.4.2 is used as a generalization of Theorem 1.4.1 for equations in one variable for finding roots in algebraic equations.

Theorem 1.4.1. (Root Search in Optimization Algorithm)
For I=[a,b] and $I \subset \mathbb{R}$, if the function $f: I \to \mathbb{R}$ is continuous, then it has at least one minima on this interval and if $\left| f(x_i) \right| = 0$ then there exists at least one $x_i \in I$, $(i \in \mathbb{N})$ satisfying this equality (Köse et al., [2]).

Theorem 1.4.2. (Root Finding Algorithm for Nonlinear Equation Systems)

Let I=[a,b] and $I \subset \mathbb{R}$, If the functions $f_i : I^n \rightarrow \mathbb{R}$ are continuous, then for each $1 \leq i \leq n$ the functions f_i have at least one minimum value in this interval and have at least one point $x = (x_1, x_2, ...x_n) \in I$ that satisfies the equality $\sum_{i=1}^{n} \left| f_i\left(x_i\right) \right| = 0$ (Köse et al., [2]).

1.5. Numerical Example

In [1], Canale and Capra considered a system of equations consisting of functions of two variables $f_1(x, y)$ and $f_2(x, y)$

$$\begin{aligned} f_1(x, y) &= x^2 + xy - 10 = 0 \\ f_2(x, y) &= y + 3xy^2 - 57 = 0 \end{aligned} \quad (2)$$

Since the real roots of this system of equations are x = 2 and y = 3, he used the fixed-point iteration method and the Newton-Raphson method to solve the system of equations, starting with initial guesses x = 1.5 and y = 3.5.

In this study, the same problem will be addressed using a mathematics-based fixed-point iteration method and a meta-heuristic, the teaching-learning algorithm. Throughout the paper, $f_1(x, y)$ and $f_2(x, y)$ will be replaced by f_1 and f_2 , respectively, in the equation system given by (2).

2. APPLICATION OF METHODS AND ALGORITHMS

In Section 1.5, the success of the approximate solution of the equation system given by (2), which consists of nonlinear equations in two variables, will be measured first by the fixed-point iteration method and then by teaching-learning algorithm.

2.1. Fixed Point Iteration Method Application

Fixed point iteration functions in two variables associated with the functions f_1 and f_2, will be considered

$$\begin{aligned} &g_1(x,y) = \frac{10}{x+y}, g_2(x,y) = \frac{57}{1+3xy}, g_3(x,y) = \frac{10}{x} - x, \\ &g_4(x,y) = \frac{57-y}{3y^2}, g_5 = \sqrt{10-xy}, g_6(x,y) = \sqrt{\frac{57-y}{3x}} \end{aligned} \quad (3)$$

These functions will be denoted as g_1, g_2, g_3, g_4, g_5, g_6 for short. In this study, we have created three different sets of iteration functions for the functions f_1 and f_2. The fixed point iteration functions related to the function f_1 are g_1, g_3, g_5 and fixed point iteration functions related to the function f_2 are g_2, g_4, g_6 . The iteration steps will be performed by taking $g_1 = x$, $g_2 = y$, $g_3 = y$, $g_4 = x$, g_5=x, $g_6 = y$ and by choosing initial conditions as $x_0 = 1,5$ and $y_0 = 3,5$. The calculations were performed for all three iteration function sets by taking the maximum number of iterations as 50 and the tolerance value as 0.01 in the MATLAB program.

Capra and Canale also discussed iteration function sets in their book as in the following forms [1].

$$g_3^*(x,y)=\frac{10-x^2}{y}=x,\ \ g_4^*(x,y)=57-3xy^2=y \tag{4}$$

$$g_5(x,y)=\sqrt{10-xy}=x,\ \ g_6(x,y)=\sqrt{\frac{57-y}{3x}}=y \tag{5}$$

In addition to the iteration sets considered by Capra and Canale, it can be seen that the iteration sets g_1 and g_2 given in (3) are also considered in this study. The iteration set given by (4) considered by Capra and Canale is in the form $g_3^*(x,y)=x, g_4^*(x,y)=y$, but in this work, unlike the previous one, $g_3=y,\ g_4=x$ is taken. These iteration steps can be practical and fast, depending on the experience of the mathematician solving the system in the normal method. But when we ask the Artificial Intelligent (AI) to generate these functions, it immediately suggests the convergent iteration function from Capra and Canale's book as the iteration function. But it does not suggest that there may be other functions and how they can be selected when a problem arises. We form the equation in mathematical theory about this. When we take the first derivative of the iteration function and set the initial condition in the first derivative, we claim that it can converge if the result is less than 1.

This example illustrates the most serious shortcoming of fixed-point iteration, namely that convergence often depends on the way the equations are formulated. Moreover, even in cases where convergence is possible, divergence can occur if the initial guesses are not close enough to the true solution. Using simple reasoning, it can be seen that sufficient conditions for convergence are of the form

$$\left|\frac{\partial f_1}{\partial x}\right|+\left|\frac{\partial f_1}{\partial y}\right|<1$$

and

$$\left|\frac{\partial f_2}{\partial x}\right|+\left|\frac{\partial f_2}{\partial y}\right|<1$$

for the case with two equations. These criteria are so restrictive that fixed point iteration can be considered of limited utility in solving nonlinear systems. However, it can be seen that the contribution of this method is greater when solving linear systems.

For each iteration function set considered in this study, the fact that $\left|g_i'(x_0)\right|=\left|\frac{\partial g_i}{\partial x}(x_0)\right|+\left|\frac{\partial g_i}{\partial y}(x_0)\right|<1$ for $1\le i\le 6$ also gives an idea about the result under the initial condition x_0.

Using the iteration function set that satisfies this condition is more appropriate to ensure convergence, otherwise a divergence from the true solution will occur. Let us now give the implementation steps of both algorithms below.

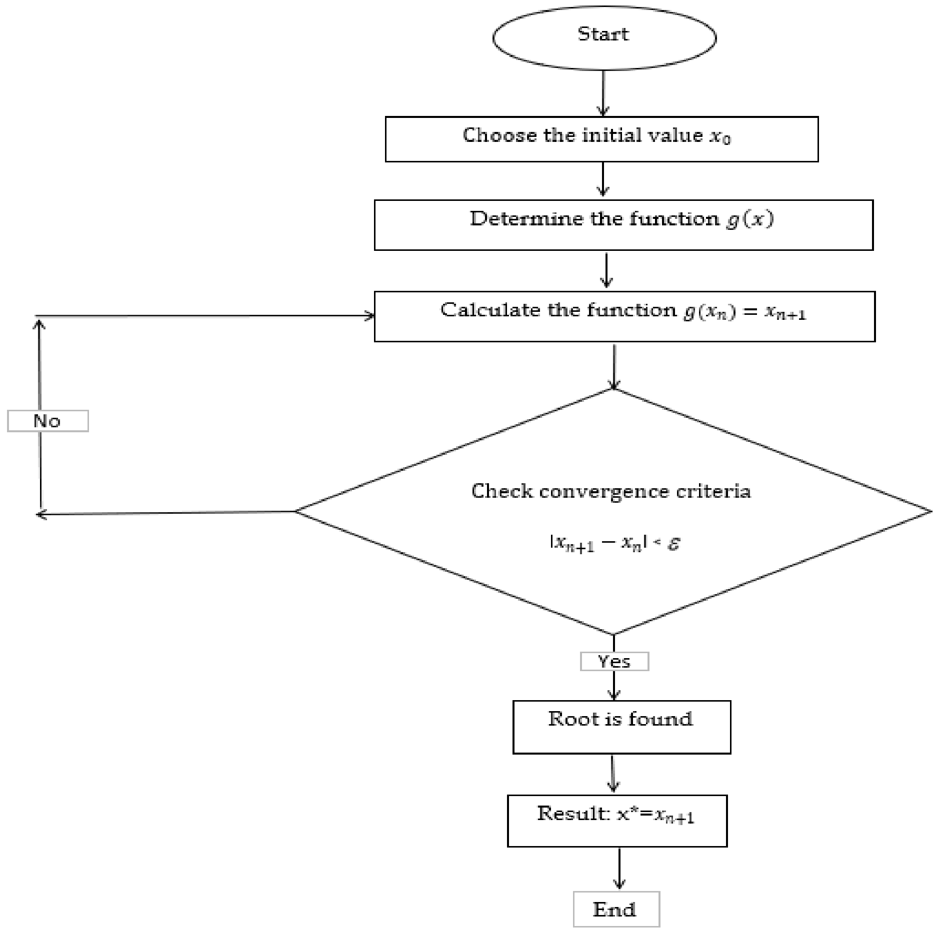

Figure 2. Basic flow diagram of fixed-point iteration method

2.2. Teaching-Learning Based Optimization Algorithm Application

In order to solve the system of equations given by (2), TLBO method is used by taking the number of populations 40, the number of variables in the population 2, the upper bound [10, 10] and the lower bound [-8, -8].

In each iteration, the best result x and y result and its value in the function are shown.

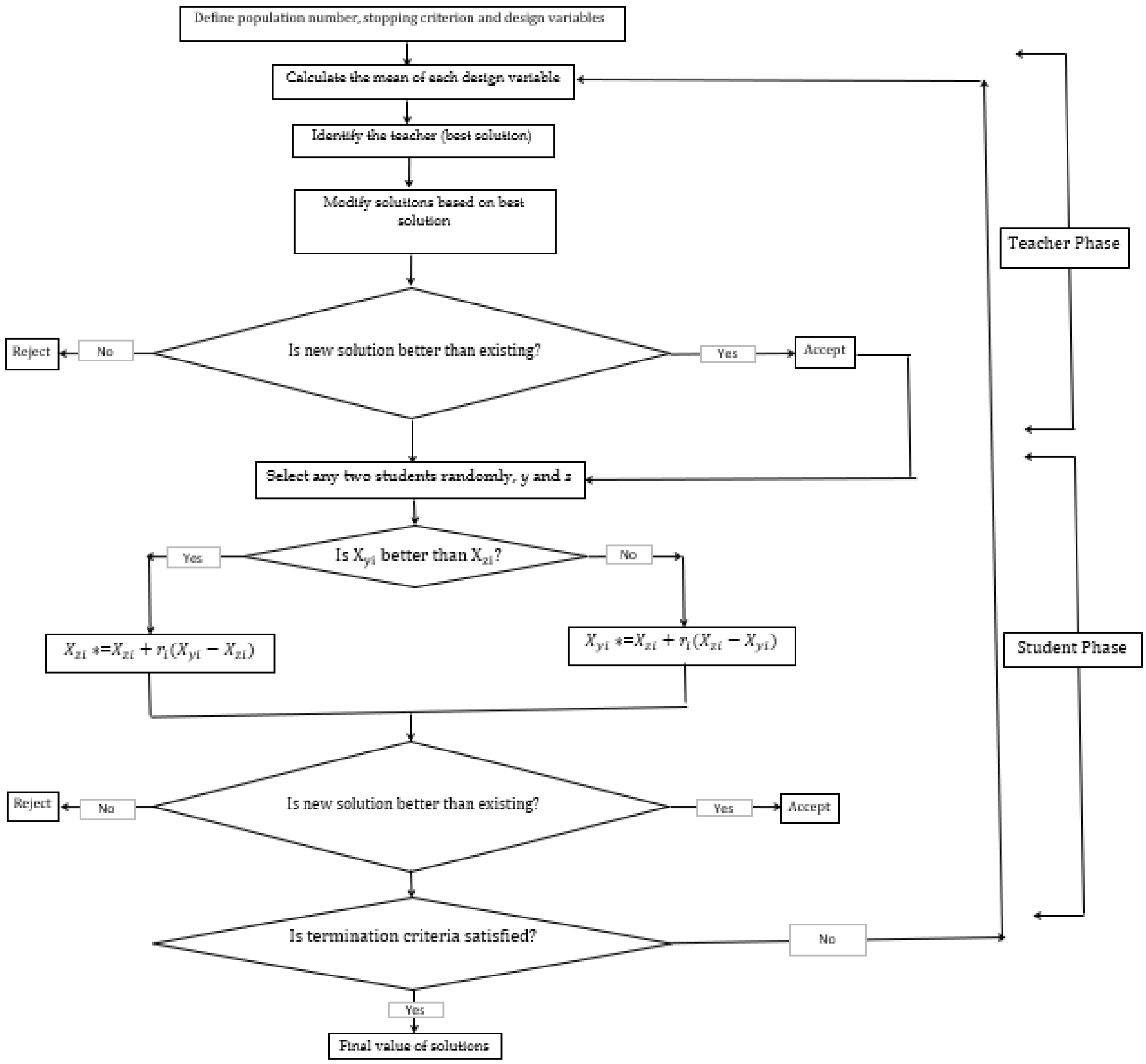

Figure 3. Flow diagram of learning-teaching based optimization algorithm

3. PERFORMANCE RESULTS

In this section, firstly, the convergence performance results of the system of equations given by (2) on three different iteration function sets obtained by the fixed-point iteration method are compared. Then, the performance results of teaching-learning based optimization algorithms for finding approximate solutions of the same equation system are obtained.

The following table, shows the convergence tables for $g_1, g_2, g_3, g_4, g_5, g_6$ fixed point iteration function sets, where *x* and *y* are solutions, the error of *x* is E_x and the error of *y* is E_y .

	Data for function set g_1, g_2				Data for function set g_3, g_4				Data for function set g_5, g_6			
Number of iterations	x	y	E_x	E_y	x	y	E_x	E_y	x	y	E_x	E_y

1	2,0000	3,4030	0,5000	0,0970	5,1667	1,4558	3,6667	2,0442	2.1794	3.4480	0,6794	0,0520
2	1,8508	2,6613	0,1492	0,7417	-3,2312	8,7362	8,3978	7,2805	1.5764	2.8619	0,6030	0,5861
3	2,2162	3,6129	0,3654	0,9515	0,1363	0.2108	3,3675	8,5255	2.3427	3.3834	0,7663	0,5215
4	1,7155	2,2781	0,5007	1,3348	73,2092	426,0335	73,0729	425,8227	1.4400	2.7620	0,9027	0,6213
5	2,5040	4,4796	0,7885	2,2015	-73,0726	-0,0007	146,2819	426,0342	2.4541	3.5433	1,0141	0,7813
6	1,4320	1,6450	1,0721	2,8346	72,9358	4,1366*10^7	146,0084	4,1366*10^7	1.1421	2.6946	1,3120	0,8487
7	3,2500	7,0662	1,8180	5,4212	-72,987	0	145,7345	4,1366*10^7	2.6311	3.9811	1,4889	1,2865
8	0,9693	0,8155	2,2806	6,2506	72,6613	2,9261*10^17	145,4600	2,9261*10^17	0	2.5917	2,6311	1,3894
9	5,6027	16,9061	4,6333	16,0906	-72,5237	-0,0001	145,1850	2,9261*10^17	3.1747	3.6281	3,2486	1,0364
10	0,4443	0,1999	5,1584	16,7062	72,3858	1,4641*10^37	144,9095	1,4641*10^37	2.4546	2.3581	0,7731	3,8440
11	15524	45,0089	15,0798	44,8090	-72,24767	0,0000	144,6335	1,4641*10^37	2.5174	2.0847	2,5549	0,3299
12	0,1652	0,0272	15,3589	44,9817	72,1092	3,6655*10^76	144,3569	3,6655*10^76	2.6063	2.4692	1,2893	0,9714
13	51981	56,2424	51,8156	56,2152	-71,9706	0,0000	144,0798	3,6655*10^76	2.2209	2.5194	1,0208	0,6176
14	0,0924	0,0065	51,8884	56,2359	71,8316	2,2975*10^155	143,8022	2,2975*10^155	2.3038	2.7040	0,8905	0,4933
15	101,1123	56,8975	101,0199	56,8910	-71,6924	0,0000	143,5240	2,2975*10^155	2.1611	2.7086	0,7196	0,5316
16	0,0632	0,0033	101,0490	56,8942	71,5529	Inf	143,2453	Inf	2.2072	2.7792	0,7261	0,4248
17	150,1736	56,9643	150,1103	56,9610	-71,4132	NaN	142,9661	NaN	2.1454	2.7723	0,6526	0,4646
18	0,0483	0,0022	150,1253	56,9621	71,2731	NaN	142,6863	NaN	2.1712	2.8047	0,6603	0,4129
19	198,0277	56,9817	197,9794	56,9795	-71,1328	NaN	142,4060	NaN	2.1392	2.7985	0,6254	0,4325
20	0,0392	0,0017	197,9885	56,9800	70,9922	NaN	142,1251	NaN	2.1543	2.8160	0,6291	0,4063
21	244,5107	56,9887	244,4715	56,9870	-70,8514	NaN	141,8436	NaN	2.1359	2.8117	0,6103	0,4162
22	0,0331	0,0014	244,4776	56,9873	70,7102	NaN	141,5616	NaN	2.1451	2.8221	0,6121	0,4016
23	289,5943	56,9923	289,5612	56,9909	-70,5688	NaN	141,2791	NaN	2.1338	2.8193	0,6011	0,4070
24	0,0288	0,0012	289,5655	56,9911	70,4271	NaN	140,9960	NaN	2.1396	2.8257	0,6020	0,3983
25	333,2890	56,9943	333,2602	56,9932	-70,2851	NaN	140,7123	NaN	2.1326	2.8239	0,5952	0,4015
26	0,0256	0,0010	333,2634	56,9933	70,14286	NaN	140,4280	NaN	2.1363	2.8280	0,5958	0,3961
27	375,6203	56,9956	375,5948	56,9946	-70,0003	NaN	140,1431	NaN	2.1318	2.8268	0,5915	0,3981
28	0,0231	0,0009	375,5973	56,9947	69,8574	NaN	139,8577	NaN	2.1341	2.8294	0,5918	0,3947
29	416,6204	56,9965	416,5973	56,9956	-69,7143	NaN	139,5717	NaN	2.1312	2.8286	0,5891	0,3960
30	0,0211	0,0008	416,5993	56,9957	69,5708	NaN	139,2851	NaN	2.1328	2.8303	0,5893	0,3937
31	456,3244	56,9971	456,3032	56,9963	-69,4271	NaN	138,9979	NaN	2.1309	2.8298	0,5875	0,3946
32	0,0195	0,0007	456,3049	56,9964	69,2831	NaN	138,7102	NaN	2.1319	2.8309	0,5876	0,3931
33	494,7685	56,9976	494,7490	56,9968	-69,1387	NaN	138,4218	NaN	2.1306	2.8306	0,5865	0,3937
34	0,0181	0,0007	494,7504	56,9969	68,9941	NaN	138,1328	NaN	2.1313	2.8313	0,5866	0,3927
35	531,9896	56,9979	531,9715	56,9972	-68,8492	NaN	137,8433	NaN	2.1305	2.8311	0,5858	0,3931
36	0,0170	0,0006	531,9727	56,9973	68,7039	NaN	137,5531	NaN	2.1309	2.8316	0,5859	0,3924
37	568,0242	56,9982	568,0073	56,9976	-68,5584	NaN	137,2623	NaN	2.1304	2.8314	0,5854	0,3927
38	0,0160	0,0006	568,0082	56,9976	68,4125	NaN	136,9709	NaN	2.1307	2.8317	0,5854	0,3923
39	602,9084	56,9984	602,8924	56,9978	-68,2663	NaN	136,6788	NaN	2.1303	2.8316	0,5851	0,3924
40	0,0152	0,0006	602,8932	56,9978	68,1198	NaN	136,3862	NaN	2.1305	2.8318	0,5851	0,3922
41	636,6774	56,9986	636,6623	56,9980	-67,9730	NaN	136,0929	NaN	2.1303	2.8318	0,5849	0,3923
42	0,0144	0,0005	636,6630	56,9980	67,8259	NaN	135,7990	NaN	2.1304	2.8319	0,5849	0,3921
43	669,3659	56,9987	669,3515	56,9982	-67,6785	NaN	135,5044	NaN	2.1303	2.8319	0,5848	0,3921
44	0,0138	0,0005	669,3521	56,9982	67,5307	NaN	135,2092	NaN	2.1303	2.8320	0,5848	0,3920
45	701,0076	56,9988	700,9938	56,9983	-67,3826	NaN	134,9134	NaN	2.1302	2.8319	0,5847	0,3921
46	0,0132	0,0005	700,9944	56,9984	67,2342	NaN	134,6169	NaN	2.1303	2.8320	0,5847	0,3920
47	731,6352	56,9989	731,6221	56,9985	-67,0855	NaN	134,3198	NaN	2.1302	2.8320	0,5846	0,3920
48	0,0127	0,0005	731,6226	56,9985	66,9364	NaN	134,0220	NaN	2.1303	2.8320	0,5846	0,3920
49	761,2809	56,9990	761,2682	56,9986	-66,7870	NaN	133,7235	NaN	2.1302	2.8320	0,5846	0,3920
50	0,0122	0,0004	761,2687	56,9986	66,6373	NaN	133,4244	NaN	2.1303	2.8320	0,5846	0,3920

Table 1. Convergence table for function sets $g_1, g_2, g_3, g_4, g_5, g_6$

It is seen from Table 1 that although the fixed-point iteration function set is close to the true root in the first iteration, it is observed that the roots and oscillate and do not converge to the true root as the number of iterations increases. It is also observed that the errors E_x and E_y increase continuously with the number of iterations.

It is also seen that the roots x and y in the fixed-point iteration function set are quite far from the true root, i.e. diverging. Even at the 16th iteration the error E_y goes to infinity.

For the fixed-point iteration functions g_5, g_6 it is seen that it converges to the true root with $x = 2{,}1302$ and $y = 2{,}8320$ values at the 47th iteration. When the number of iterations is further increased in this step, it can be seen that it will get closer to the true root.

Now let us present the performance results obtained with the TLBO algorithm in the table below.

Iteration Number	**Best Result** *x*	**Best Result** *y*	$f_{\min} = \lvert f_1 \rvert + \lvert f_2 \rvert$	**Iteration Number**	**Best Result** *x*	**Best Result** *y*	$f_{\min} = \lvert f_1 \rvert + \lvert f_2 \rvert$
1	5,5081	-1,8522	12,3035	**26**	1,9908	3,0072	0,0657
2	3,0623	2,4998	9,9395	**27**	1,9908	3,0072	0,0657
3	0,8620	4,4977	5,5669	**28**	1,9908	3,0072	0,0657
4	1,6075	3,2514	4,9562	**29**	1,9908	3,0072	0,0657
5	1,6075	3,2514	4,9562	**30**	1,9908	3,0072	0,0657
6	2,2156	2,7944	3,4030	**31**	1,9908	3,0072	0,0657
7	2,2156	2,7944	3,4030	**32**	1,9908	3,0072	0,0657
8	1,9962	3,0389	1,3935	**33**	1,9908	3,0072	0,0657
9	1,9962	3,0389	1,3935	**34**	1,9908	3,0072	0,0657
10	1,9978	2,9936	0,3242	**35**	1,9908	3,0072	0,0657
11	1,9978	2,9936	0,3242	**36**	1,9908	3,0072	0,0657
12	2,0144	2,9876	0,1481	**37**	1,9908	3,0072	0,0657
13	2,0144	2,9876	0,1481	**38**	2,0042	2,9963	0,0458
14	2,0144	2,9876	0,1481	**39**	2,0036	2,9975	0,0268
15	2,0144	2,9876	0,1481	**40**	2,0036	2,9975	0,0268
16	2,0144	2,9876	0,1481	**41**	2,0036	2,9975	0,0268
17	2,0144	2,9876	0,1481	**42**	2,0026	2,9981	0,0150
18	2,0144	2,9876	0,1481	**43**	1,9982	3,0013	0,0109
19	2,0144	2,9876	0,1481	**44**	1,9982	3,0013	0,0109
20	2,0144	2,9876	0,1481	**45**	1,9982	3,0013	0,0109
21	2,0144	2,9876	0,1481	**46**	1,9982	3,0013	0,0109
22	2,0142	2,9882	0,1305	**47**	1,9982	3,0013	0,0109
23	2,0142	2,9882	0,1305	**48**	1,9982	3,0013	0,0109
24	2,0142	2,9882	0,1305	**49**	1,9995	3,0005	0,0072
25	2,0142	2,9882	0,1305	**50**	1,9995	3,0005	0,0072

Table 2. Convergence data with TLBO algorithm

It can be seen from the table above that the best solution with the TLBO algorithm is found as $f_{\min} = 0{,}0072$ at the 49th iteration with *x=1.9995* and *y=3.0005*. This shows that the TLBO algorithm can be used as a successful approach that is very close to the real solution.

3.1. Convergence Graphs

The graphs showing the convergence speed of both algorithms in the root finding process according to the number of iterations are given below.

3.1.1. Convergence Graphs for Fixed Point Iteration Function Sets

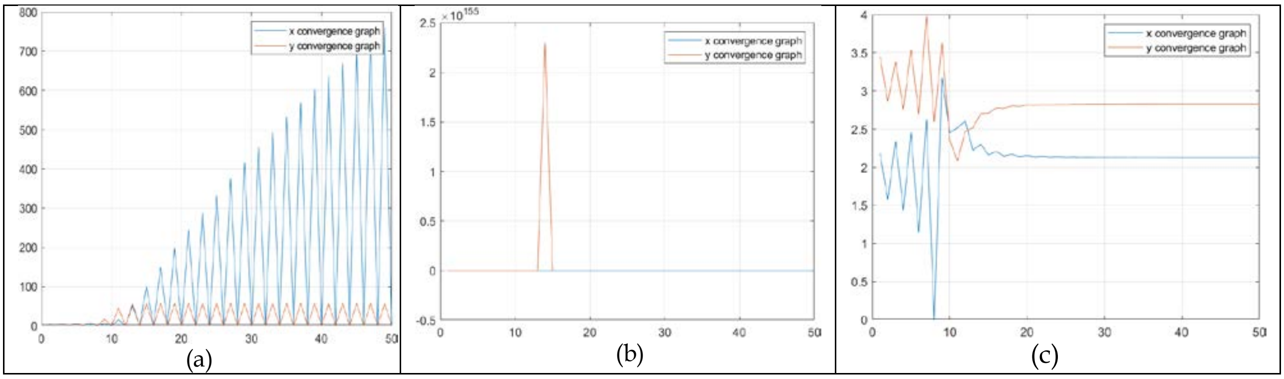

Figure 4. (a) Convergence graphs of fixed-point iteration set $g_1 = x, g_2 = y$
(b) Convergence graphs of fixed-point iteration set $g_3 = y, g_4 = x$
(c) Convergence graphs of fixed-point iteration set $g_5 = x, g_6 = y$

Figure 4 (a) shows that in the fixed-point iteration function set $x = g_1,\ y = g_2$, variable x converges to 1.25 but does not converge to the true root 2, variable y converges to 0.9 but does not converge to the true root 3. Therefore, $x = g_1,\ y = g_2$ fixed point iteration function set is divergent to the true root.

Figure 4 (b) shows that the x and y roots oscillate and do not converge to the true root in the fixed-point iteration function set g_3, g_4 .

Figure 4 (c) shows that in the $g_5 = x, g_6 = y$ fixed point iteration function set, x converges to 2.1 and y converges to 2.8, and as the number of iterations increases, it converges to the true root x=2 and y=3.

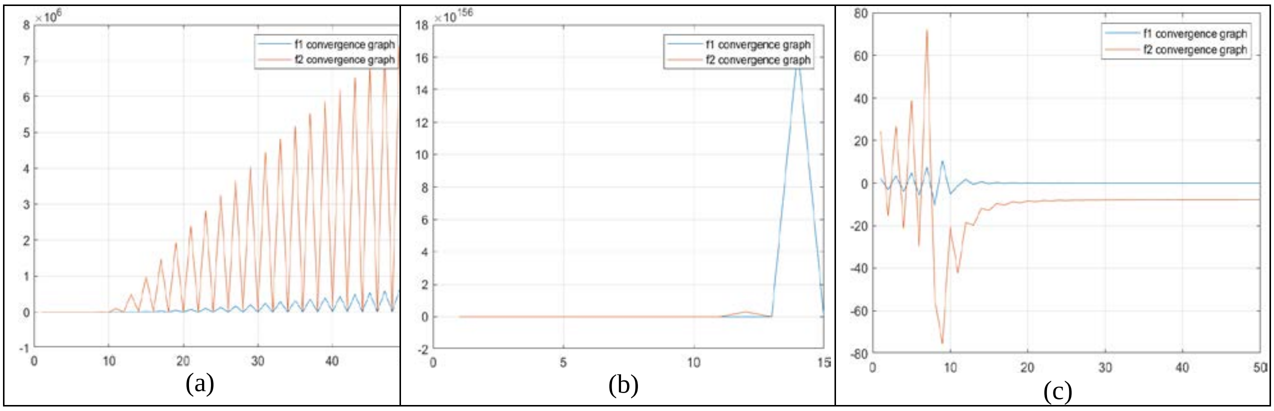

Figure 5. (a) Convergence graph for fixed point iteration function sets $x = g_1,\ y = g_2$ of functions f_1 ve f_2
(b) Convergence graph for fixed point iteration function sets $x = g_4,\ y = g_3$ of functions f_1 ve f_2
(c) Convergence graph for fixed point iteration function sets $x = g_5,\ y = g_6$ of functions f_1 ve f_2

According to Figure 5 (a) and (b), it can be seen that the functions f_1 and f_2 do not converge to zero for the values $x = g_1,\ y = g_2$ and $x = g_4,\ y = g_3$ obtained from the fixed point iteration function sets and therefore diverges. Figure 5 (c) shows that the functions of f_1 and f_2 converge to zero for the fixed-point iteration function set $x = g_5,\ y = g_6$. When the x and y values obtained from the functions $x = g_5,\ y = g_6$ are substituted into the functions f_1 and f_2 , f_1 approaches zero, that is, the true root, while f_2 approaches a value close to zero.

In order to find the root with Heuristic Optimization algorithms, the convergence to the root is checked by taking the sum of the absolute values of the objective functions using Theorem 1.4.2. Therefore, there is no need for derivatives or generating extra functions to solve the root finding problem. The objective function must converge to the minimum value. Here, the convergence to zero of the sum of the absolute values of the objective functions for the approximate roots of the fixed point iteration and the approximate roots of the heuristic method is graphically compared and given in Figures 6 and 7.

As a result of Theorem 1.4.2, now let us present the graphs obtained from the absolute sums of f_1 and f_2 values in Figure 5. The approach of the sum of $|f_1|+|f_2|$ to zero depending on the iteration number is given in Figure 6.

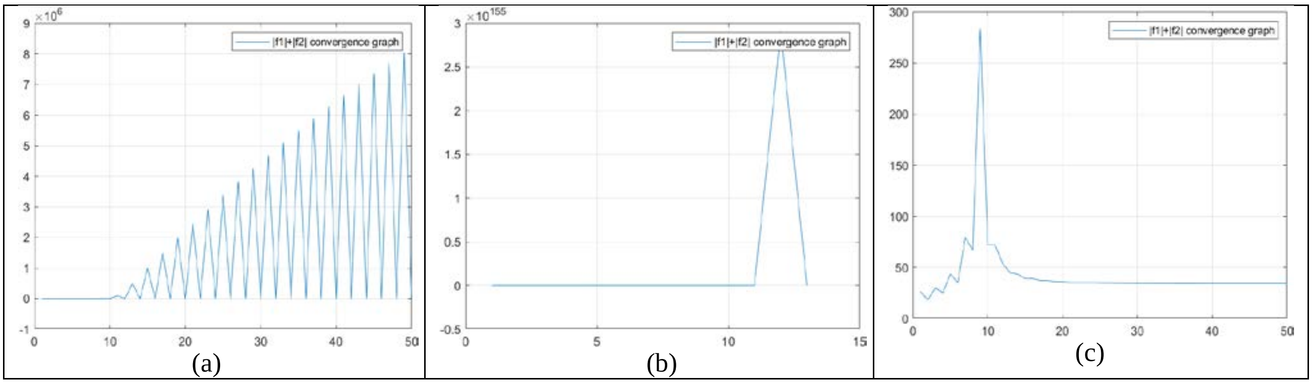

Figure 6. (a) Convergence graphs of fixed-point iteration functions $x = g_1,\ y = g_2$ of functions $|f_1|+|f_2|$

(b) Convergence graphs of fixed-point iteration functions $y = g_3,\ x = g_4$ of functions $|f_1|+|f_2|$

(c) Convergence graphs for fixed point iteration functions $x = g_5,\ y = g_6$ of functions $|f_1|+|f_2|$

In Figure 6 (a), it can be seen that the sum of $|f_1|+|f_2|$ for the first two selected iteration formulas does not approach zero and makes a fluctuating search for fixed point iteration function sets $x = g_1,\ y = g_2$. Figure 6 (b) shows that the sum of $|f_1|+|f_2|$ does not converge to zero for fixed point iteration function sets $x = g_4,\ y = g_3$, while Figure 6 (c) shows that the sum of $|f_1|+|f_2|$ converges to zero for fixed point iteration functions $x = g_5,\ y = g_6$.

3.1.2. Convergence Graphs for the Teaching-Learning Algorithm

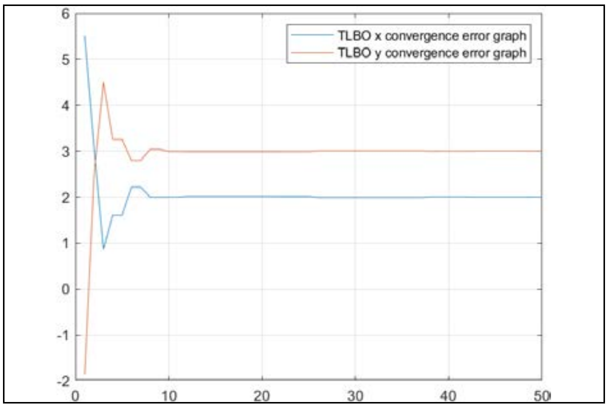

Figure 7. Convergence error graph of TLBO algorithm

According to Figure 8, it is seen that the error margins in x and ye values are very close to 0.01 until the 10th iteration with the TLBO algorithm. In this respect, it can be said that the TLBO algorithm approaches the actual x and y values with very little error.

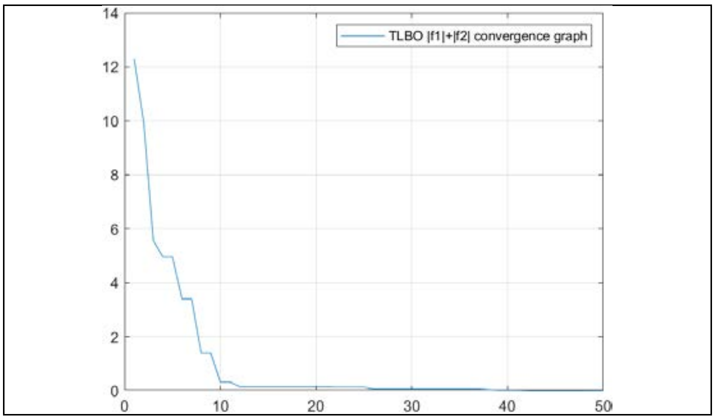

Figure 8. Zero convergence graph of $f_{\min} = |f_1| + |f_2|$ for TLBO algorithm.

Figure 8 shows that the value of $f_{\min} = |f_1| + |f_2|$ starts to approach zero after the 10th iteration and reaches its closest value to zero at the 49th iteration.

3.2. Error Analysis Graphs

The graphs comparing the error values at each iteration step for the fixed-point iteration method and the teaching-learning based optimization methods that we used to find approximate solutions of the system of equations given by (2) will be given below.

3.2.1 Convergence Error Graphs for Fixed Point Iteration Functions

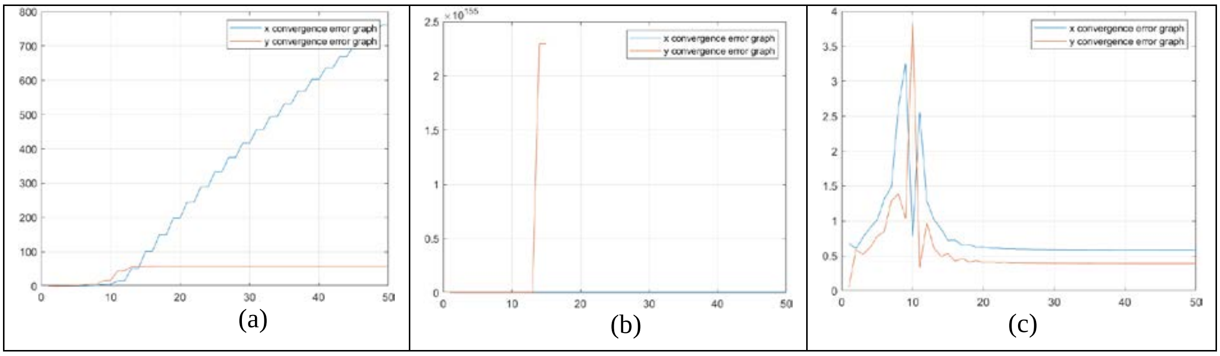

Figure 9. (a) Convergence error graphs of fixed-point iteration function set $x = g_1,\ y = g_2$
(b) Convergence error graphs of fixed-point iteration function set $x = g_4,\ y = g_3$
(c) Convergence error graphs of fixed-point iteration function set $x = g_5, y = g_6$

The convergence error graphs of fixed-point iteration functions are presented in Figure 9. Figure 9 (a) and Figure 9 (b) illustrate that the errors obtained for the fixed-point iteration functions exhibit fluctuations, whereas Figure 9 (c) demonstrates that the errors for the fixed-point iteration functions converge to zero.

Convergence Error Graphs for the TLBO algorithm are given in Figure 10. As illustrated in Figure 10, the convergence error graph for the TLBO algorithm exhibits a similar pattern to the graph (Figure 8) of the function converging to the minimum value. The TLBO algorithm demonstrates a consistent reduction in the approximation error as it approaches the true root. To summarize; the convergence error graphs for the TLBO algorithm, display a consistent reduction in approximation error, similar to the function's graph in Figure 8, as the algorithm steadily converges to the true root for variables (x) and (y).

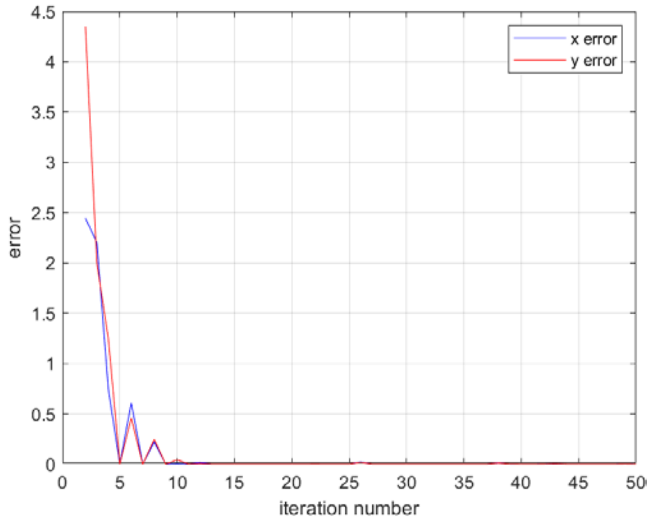

Figure 10. Convergence error graphs of the TLBO algorithm x, y

4. CONCLUSION

It is seen that TLBO converges better to the true root according to the number of iterations. Since the fixed-point iteration method aims to approach the best solution by considering different iteration sets, the success to be achieved here varies according to the choice of iteration sets. Even from this point of view, the fixed-point iteration method is an optimization method that requires more operations and cannot be said to be more successful than the TLBO algorithm in terms of convergence in the problem considered.

Since the problem considered in this paper consists of only two nonlinear equations, the analysis of computation times does not make a significant difference, since current computers are quite powerful and therefore the total computation times of the algorithms differ by milliseconds. In more complex systems, with more equations, the time difference can be more discriminating.

Choosing different functions can lead to better results, but there's a risk of non-convergence due to dependency on function creation and initial values. The functions from Canale and Capra's book [1] are used here as they are standard references, helping those interested in the field to understand the topic and make comparisons.

Heuristic optimization techniques, like numeric methods, don't provide exact solutions but can get close to the real solution. By setting a maximum number of iterations or acceptable error margins, we can achieve a good approximation. In the teaching and learning algorithm, iterations are capped at 50 steps to avoid repetition, usually resulting in a stable approximation despite further iterations.

5. ACKNOWLEDGEMENTS

The authors would like to thank the reviewers and editors of Journal of Universal Mathematics.

Funding
The authors declared that has not received any financial support for the research, authorship or publication of this study.

The Declaration of Conflict of Interest/ Common Interest
The authors declared that no conflict of interest or common interest.

The Declaration of Ethics Committee Approval
This study does not be necessary ethical committee permission or any special permission.

The Declaration of Research and Publication Ethics
The authors declared that they comply with the scientific, ethical, and citation rules of Journal of Universal Mathematics in all processes of the study and that they do not make any falsification on the data collected. Besides, the author declared that Journal of Universal Mathematics and its editorial board have no responsibility for any ethical violations that may be encountered, and this study has not been evaluated in any academic publication environment other than Journal of Universal Mathematics.

REFERENCES

[1] R. P. Chapra, S.C. Canale, Numerical Methods for Engineers, McGraw-Hill Higher Education, New York, (2002).
[2] B. Köse, B. Demirtürk, and Ş. Konca, Finding Solutions of Nonlinear Equation Systems with Newton Raphson and Red Fox Methods, 6th International Graduate Studies Congress-IGSCONG 2024, June 5-8, 2024.
[3] B. Demirtürk, B. Köse, Ş. Konca, Finding the Solution of a System of Nonlinear Equations with Sine Cosine and Particle Swarm Optimization Algorithms, 6th International Istanbul Modern Scientific Research Congress, 5-7 July 2024, Istanbul, (2024).
[4] T. Gemechu, S. Thota, On New Root Finding Algorithm for Solving Nonlinear Transcendental Equations, International Journal of Chemistry, Mathematics and Physics (IJCMP), Vol.4, No.2, pp.18-24 (2020).
[5] T. Gemechu, Root finding for nonlinear equations, Mathematical Theory and Modeling, Vol.8, No.7, 10 pages, (2018).
[6] V. Y. Semenov, A Method to Find all the Roots of the System of Nonlinear Algebraic Equations Based on the Krawczyk Operator, Cybernetic and Systems Analysis, Vol.51, No.5, pp.819-825 (2015).
[7] J. M. Ortega, W. C. Rheinbolt, Iterative Solution of Nonlinear Equations in Several Variables; Academic Press: New York, NY, USA, (1970).
[8] C. T. Kelley, Iterative methods for linear and nonlinear equations, Society for Industrial and Applied Mathematics, (1995).
[9] J. F. Traub, Iterative Methods for the Solution of Equations; American Mathematical Soc.: Washington, WA, USA, (1982).
[10] C. G. Broyden, A class of methods for solving nonlinear simultaneous equations, Mathematics of Computation, Vol.19, pp.577-593 (1965).
[11] B. Köse, F. Kaya, Öğretme ve Öğrenme Tabanlı Optimizasyon, Teknobilim 2023: Optimizasyon Modelleme ve Yapay Zeka Optimizasyon Algoritmaları, Efe Akademi, İstanbul, (2023).
[12] R.V. Rao, V. J. Savsani and D. P. Vakharia, Teaching-Learning-Based Optimization: A Novel Method for Constrained Mechanical Design Optimization Problems. Computer-Aided Design, Vol.43, pp.303-315 (2011).
[13] K. Ömeroğlu, Lineer Olmayan Denklemler ve Geogebra Uygulamaları, Recep Tayyip Erdoğan Üniversitesi Fen Bilimleri Enstitüsü Yüksek Lisans Tezi, (2019).

IZMIR BAKIRÇAY UNIVERSITY, DEPARTMENT OF FUNDAMENTAL SCIENCES, FACULTY OF ENGINEERING AND ARCHITECTURE, İZMİR, TURKEY
E-mail address: bahar.demirturk@bakircay.edu.tr

IZMIR BAKIRÇAY UNIVERSITY, DEPARTMENT OF ELECTRIC ELECTRONIC ENGINEERING, FACULTY OF ENGINEERING AND ARCHITECTURE, İZMİR, TURKEY
E-mail address: bayram.kose@bakircay.edu.tr

IZMIR BAKIRÇAY UNIVERSITY, DEPARTMENT OF FUNDAMENTAL SCIENCES, FACULTY OF ENGINEERING AND ARCHITECTURE, İZMİR, TURKEY
E-mail address: sukran.konca@bakircay.edu.tr