SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
DERGİSİ

Sakarya University Journal of Science
SAUJS

ISSN 1301-4048 e-ISSN 2147-835X Period Bimonthly Founded 1997 Publisher Sakarya University
http://www.saujs.sakarya.edu.tr/

Title: Enhanced Tunicate Swarm Algorithm For Big Data Optimization

Authors: Emine BAŞ

Recieved: 2022-10-27 00:00:00

Accepted: 2023-01-19 00:00:00

Article Type: Research Article

Volume: 27
Issue: 2
Month: April
Year: 2023
Pages: 313-334

How to cite
Emine BAŞ; (2023), Enhanced Tunicate Swarm Algorithm For Big Data Optimization .
Sakarya University Journal of Science, 27(2), 313-334, DOI:
10.16984/saufenbilder.1195700
Access link
https://dergipark.org.tr/en/pub/saufenbilder/issue/76551/1195700

Sakarya University Journal of Science 27(2), 313-334, 2023

Enhanced Tunicate Swarm Algorithm for Big Data Optimization

Emine BAŞ *1

Abstract

Today, with the increasing use of technology tools in daily life, big data has gained even more importance. In recent years, many methods have been used to interpret big data. One of them is metaheuristic algorithms. Meta-heuristic methods, which have been used by very few researchers yet, have become increasingly common. In this study, Tunicate Swarm Algorithm (TSA), which has been newly developed in recent years, was chosen to solve big data optimization problems. The Enhanced TSA (ETSA) was obtained by first developing the swarm action of the TSA. In order to show the achievements of TSA and ETSA, various classical benchmark functions were determined from the literature. The success of ETSA has been proven on these benchmark functions. Then, the successes of TSA and ETSA are shown in detail on big datasets containing six different EEG signal data, with five different population sizes (10, 20, 30, 50, 100) and three different stopping criteria (300, 500, 1000). The results were compared with the Jaya, SOA, and SMA algorithms selected from the literature, and the success of ETSA was determined. The results show that ETSA has sufficient success in solving big data optimization problems and continuous optimization problems.

Keywords: TSA, tunicate, meta-heuristic, big data

1. INTRODUCTION

Various institutions and organizations have recently emphasized the importance of collecting data from users, customers or participants and using this data to make various decisions [1]. The greater the amount of data collected, the higher the accuracy of the operations performed on the data. With the widespread use of technology tools, huge amounts of data are growing, including hundreds, sometimes even thousands, of variables. Thus, the importance of analyzing various sources with their outputs in a fast and short time increases [1]. Such data has frequently grown to sizes that conventional analytical techniques cannot handle. As a result, a new term, big data, has been introduced into the computational and information sciences literature to describe such data [2-4]. When a data chunk is considered big data, its 4V characteristics (volume, velocity, diversity, accuracy) should be examined. The volume property is used to describe how large the amount of data is. The speed feature represents the data collection rate. The diversity of resources is expressed by the diversity feature. Accuracy refers to the quality of data sources data [2-4]. They also

* Corresponding author: ebas@ktun.edu.tr (E.BAŞ)
[1] Konya Technical University
ORCID: https://orcid.org/0000-0003-4322-6010

contain a large number of decision variables, objective functions, or functions with mathematical properties, such as variables, and sometimes require real-time solutions. One of today's problems is to analyze big data and make it meaningful. It is almost impossible to examine these data in a reasonable time with classical mathematical methods. Therefore, researchers have started to search for different methods to analyze and interpret this huge data faster. One of the methods they found was heuristic algorithms.

Many researchers have expressed interest in a recent big data optimization problem introduced by Goh et al., which has been investigated using a few meta-heuristic algorithms. They separated the EEG signals into two sections. The first part is similar to the source signal in that it provides the necessary information. The second part is accompanied by artifacts or noise [5]. EEG signal data are grouped into six different data sets named EE4, EE4N, EE12, EE12N, EE19, and EE19N, with reference to the number of interconnected time series. Each time series has a length of 256. While the EE4 problem example has four interconnected time series, the EE12 and EE19 problem examples have twelve and nineteen interconnected time series, respectively. There have been many studies in the literature that have solved the mentioned problem. Zhang et al. pioneered the literature by solving the aforementioned problem with metaheuristics. They solved the aforementioned problem with the variant they created by using crossover and mutation operators named MOME/D [6]. Elsayed and Sarker proposed the ADEF approach, which is based on a DE algorithm, and solved the related problem [7]. Majdouli et al. used the Fireworks algorithm (FW) to solve big data optimization problems [8]. Cao et al. proposed and solved big data optimization problems using a new metaheuristic called Phase Based Optimization (PBO) [9]. Aslan and Karaboga tested the success of the genetic Artificial Bee Colony algorithm in a big data optimization problem [1].

Kaur et al. proposed a new bio-inspired meta-heuristic algorithm [10]. Tunicate Swarm Algorithm (TSA) was inspired by sea creatures called Tunicates. Tunicates are marine creatures capable of locating food sources in the seas. Tunicates rely on two key behaviors in navigation and foraging for food: jet propulsion and swarm intelligence. Thanks to these two movements, it performs exploration and exploitation operations. Algorithms with swarm behavior are successful to the extent that they balance these two abilities well. In this study, herd behavior has been updated and Enhanced TSA (ETSA) has been proposed to increase the success of TSA. In TSA, only the individual with the best resource was taken into account and the population was searched locally, while in ETSA, the individual with the worst resource was also considered. Thus, it is planned to increase the success of TSA. The balance between exploration and exploitation is better maintained.

Two sets of two problems were used to test the success of TSA and ETSA. In the first problem set, 23 unimodal, multimodal and fixed-dimensional multi-modal test functions were selected from the literature. Descriptions of these test functions are provided in the appendix. In these test functions of TSA and ETSA, parameter analysis was performed for the population sizes (10, 20, 30, 50, and 100), and the most appropriate population size was determined as 30. The performances of TSA and ETSA were tested in detail for population size=30, maximum iteration=500, and dimension=30 on 23 test functions. The results were shown according to the standard deviation, mean, best, and time comparison criteria for TSA and ETSA. The convergence graphs were shown. ETSA achieved better results than TSA in all benchmark functions except F8 and F20. In order to prove the success of ETSA, the results were compared with various heuristic algorithms selected from the literature in recent years. These algorithms are Jaya Algorithm [11], Skill Optimization Algorithm (SOA) [12], and Slime Mould Algorithm (SMA) [13]. According to the results, ETSA achieved better results than Jaya. ETSA was the most successful algorithm after SMA and SOA. It was proved by Wilcoxon signed test that there was a significant difference on the results.

In the second problem, big datasets containing six different EEG signals (EE4, EE4N, EE12, EE12N, EE19, and EE19N) were selected from the literature. The performances of TSA and ETSA were tested in detail for five different population sizes (10, 20, 30, 50, and 100) and three different maximum iterations (300, 500, 1000) on big datasets containing six different EEG signals. These datasets contain large dimensions (EE4 and EE4N for 1024, EE12 and EE12N for 3072, EE19 and EE19N for 4868). That is why it is called the big data optimization problem. The results were tested according to the standard deviation, mean, best, and time comparison criteria for TSA and ETSA, and convergence graphs were shown. The results have been compared. TSA and ETSA results were compared with new metaheuristic algorithms (Jaya, SOA, and SMA) selected from the literature and developed in recent years, and their success was demonstrated. Convergence graphs were drawn. According to the results, ETSA achieved better results than TSA. Thus, it has been proven that In terms of working time, TSA worked in a shorter time than ETSA. The reason for this is the codes added for the development of ETSA. It is not a significant time difference. When compared with various algorithms selected from the literature, it achieved the best results after SOA. The results obtained were subjected to the Wilcoxon sign test, which is a statistical test, and it was shown that there was a semantic difference between the results obtained.

The results showed that ETSA achieved better results than TSA and showed preferable success for continuous optimization problems. The motivations and contributions of this study can be summarized as follows:

- ETSA was proposed for the first time in this study by improving the local search capability of TSA. In ETSA, not only the best source information, but also the worst source information was used to change the position of the population.
- TSA and ETSA were tested on two different problem sets (a-) twenty-three classic test functions, b-) big data optimization problem). The results showed that ETSA achieved better results than TSA. The success of ETSA was proven by applying the Wilcoxon signed test on the results.
- For the big data optimization problem, TSA and ETSA were tested for the first time in this study and their results were presented to the literature.
- The performances of TSA and ETSA were tested in detail for five different population sizes (10, 20, 30, 50, and 100) and three different maximum iterations (300, 500, 1000) on big datasets containing six different EEG signals.
- The success of ETSA has been compared with various heuristic algorithms (Jaya, SOA, and SMA) selected from the literature and proposed in recent years. ETSA managed to enter the top three in the classical test functions and the top two in the big data optimization problem.

The remainder of this work is structured as follows: TSA is explained in detail in Section 2. ETSA is detailed in Section 3. In Section 4, the big data optimization problem is defined. In Section 5, performance tests of

TSA and ETSA algorithms on classical 23 test functions and on big data optimization problem datasets are performed and compared with each other. Convergence graphs were drawn.

2. MATERIALS AND METHODS

2.1. Standard Tunicate Swarm Algorithm

The Tunicate Swarm Algorithm (TSA) is a recent bio-inspired meta-heuristic algorithm. TSA was inspired by tunicates. Tunicates are marine creatures capable of locating food sources in the seas. Tunicates navigate and forage for food using two key behaviors: jet propulsion and swarm intelligence. In the mathematical model of jet propulsion, a tunicate moves towards the position of the best population individual and stays close to the best population individual, while avoiding conflicts between population members. In herd behavior, population members update their positions according to the best population individual [10].

Mathematical Model of Jet Propulsion:

This behavior was developed to ensure the social balance of power among tunicates and to prevent collisions between them. It is shown by Equations 1-5.

$$\vec{A} = \frac{\vec{G}}{\vec{M}} \tag{1}$$

$$\vec{G} = r_2 + r_3 - \vec{F} \tag{2}$$

$$\vec{F} = 2.r_1 \tag{3}$$

$$\vec{M} = \lfloor X_{min} + r_1.X_{max} - X_{min} \rfloor \tag{4}$$

X_{min} and X_{max} values are taken as 1 and 4, respectively. Necessary parameter analyzes were made by Kaur et al. [10]. In this study, a parameter analysis was not performed for these values.
The movement of the population members towards the best neighbor direction is shown by Equation 5.

$$\overrightarrow{T_{distance}} = \left|\overrightarrow{T_{source}} - r.\overrightarrow{T(x)}\right| \tag{5}$$

where $\overrightarrow{T_{distance}}$ represents the distance between the food source and the population individual, and x represents the current iteration, $\overrightarrow{T_{source}}$ is the location of food source (the location of best tunicate), $\overrightarrow{T(x)}$ indicates the location of tunicate and r is a random number between [0, 1].

$$\overrightarrow{T(x')} = \begin{cases} \overrightarrow{T_{source}} + \vec{A}.\overrightarrow{T_{distance}}, & if\ r \geq 0.5 \\ \overrightarrow{T_{source}} - \vec{A}.\overrightarrow{T_{distance}}, & if\ r < 0.5 \end{cases} \tag{6}$$

where $\overrightarrow{T(x')}$ is the updated $\overrightarrow{T(x)}$ (location of tunicate) with respect to $\overrightarrow{T_{source}}$ (the location of food source) and r is a random number between [0, 1].

Mathematical Model of Swarm Behavior

The herding behavior of a tunicate is illustrated by the following equation:

$$T(\vec{x}+1) = \frac{\overrightarrow{T(x')} + T(\vec{x}-1)}{2+r1} \tag{7}$$

Algorithm 1 depicts the TSA algorithm's pseudocode.

Algorithm 1: Tunicate Swarm Algorithm

```
Input:  Tunicate population T_i (i=1, 2, 3, ..., pop)
Output: Optimal tunicate individual
Procedure: TSA
Initialize parameters X_min, X_max, etc.
Calculate the fitness value of each tunicate
T_source    identify the best tunicate individual
  While (iteration< Mak_iterations ) do
    for i=1 to pop do
      Update the position of each tunicate individual using Eq. (7).
    end for
    Update parameters (A, G, F, and M)
    Check tunicate populations
    Update T_i if there is a better solution than the previous optimal solution
    iteration <- iteration +1
  end while
Return T_i
end procedure
```

Algorithm 1 can be detailed as follows:

Step 1: The procedure for the TSA is created.

Step 2: The initial parameters for TSA are defined.

Step 3: The fitness function is calculated for each individual of the tunicate population.

Step 4: The population individual with the minimum value is determined from the fitness values obtained. This individual is registered as a $\overrightarrow{T_{source}}$.

Step 5: A while loop is set up for the maximum iteration cycle (steps 5-13). The iteration variable loops from zero to the maximum iteration.

Steps 6-8: Position update was performed for each individual of the tunicate population using Equation 7. Thus, new tunicate candidates are created.

Step 9: The parameter settings used in TSA ($\vec{A}$, $\vec{G}$, $\vec{F}$, $\vec{M}$) have been updated using Equations 1-4.

Step 10: Tunicate population individuals are checked to see if they exceed the search space limits.

Step 11: The solutions of the new tunicate candidates are compared with solutions of the existing turnicate individuals. Better new individuals are updated to replace old individuals.

Step 12: The iteration variable is incremented by one.

Step 13: while loop end

Step 14: Tunicate population is returned.

Step 15: end of procedure

2.2. Enhanced Standard Tunicate Swarm Algorithm

In the standard TSA algorithm, only the positions of the best-tunicate are considered in behaviors of the jet propulsion and swarm intelligence. The position of the worst-tunicate individual was not taken into account. In the Enhanced TSA (ETSA) algorithm, the worst tunicate position was taken into account in behaviors of the jet propulsion and swarm intelligence. Equations 5 - 7 have been updated again as Equations 8 - 10. While updating tunicates' new positions, tunicate individuals try to move away from the worst search agent while approaching the best search agent, which provides further development of local search capabilities in the search space. The ETSA and TSA working steps are the same, except for the updated equations. The flowchart of ETSA is shown in Figure 1.

$$\overrightarrow{T_{dıstance\_w}} = \left|\overrightarrow{T_{worst}} - r.\overrightarrow{T(x)}\right| \tag{8}$$

$$\overrightarrow{T(x'')} = \begin{cases} \overrightarrow{T_{worst}} + \vec{A}.\overrightarrow{T_{dıstance\_w}}, & if\ r \geq 0.5 \\ \overrightarrow{T_{worst}} - \vec{A}.\overrightarrow{T_{dıstance\_w}}, & if\ r < 0.5 \end{cases} \tag{9}$$

where $\overrightarrow{T(x')}$ is the updated $\overrightarrow{T(x)}$ (location of tunicate) with respect to $\overrightarrow{T_{worst}}$ (the location of worst tunicate) and r is a random number between [0, 1].

$$T(\vec{x}+1) = \frac{\overrightarrow{T(x\prime)} + T(\vec{x}-1) + \overrightarrow{T(x\prime\prime)}}{2+r1} \tag{10}$$

2.3. Definitions of Big Data Problems on the EEG datasets

Because of the unique characteristics of big data, such as volume, speed, variety, and accuracy, big data optimization problems differ from classical optimization problems. They also contain a large number of decision variables, objective functions, or functions with varying mathematical properties, and they occasionally necessitate real-time solutions. A major data optimization problem have recently emerged, attracting the attention of researchers who wish to solve it using swarm intelligence-based and evolutionary algorithms. Big data has specific properties such as volume, speed, variety and accuracy, so the big data optimization problem is different from the classical optimization problems. They also contain a large number of decision variables, objective functions, or functions with mathematical properties, such as variables, and sometimes require real-time solutions. In recent years, Goh et al. have defined a big data optimization problem and tried to solve this problem with swarm intelligence-based evolutionary algorithms.

EEG signals were collected by Goh et al. for big data optimization problems [5, 11]. EEG signal data are grouped into six different data sets named EE4, EE4N, EE12, EE12N, EE19, and EE19N, with reference to the number of interconnected time series. Each time series has a length of 256. While the EE4 problem example has four interconnected time series, the EE12 and EE19 problem examples have twelve and nineteen interconnected time series, respectively. For problem examples EE4N, EE12N, and EE19N, there are also four, twelve, and nineteen-time series. However, as the name implies, they have been slightly modified with the addition of an extra noise component [5, 14, 15].
Suppose D is a matrix of size $K \times L$ representing the transformed problem example. For matrix E, K corresponds to the number of time series and L is equal to the length of each time series. Suppose E is a matrix of size $K \times K$ representing the transformed problem example. The problem definition is shown in Equations 11-17.

$$X = E \times D \tag{11}$$

$$D = D_1 + D_2 \tag{12}$$

$$X = E \times D_1 + E \times D_2 \tag{13}$$

$$C = \frac{covar(E,\ E \times D_1)}{var(E) \times var(E \times D_1)} \tag{14}$$

$$Minimize \quad f_1 = \frac{1}{(K^2-K)} \sum_{i\neq j}\left(C_{ij}\right)^2 + \frac{1}{K}\sum_i\left(1-C_{ij}\right)^2 \tag{15}$$

$$Minimize \quad f_2 = \frac{1}{K\times L} \sum_{ij}\left(D_{ij} - D1_{ij}\right)^2 \tag{16}$$

$$\begin{aligned} &Minimize && f_1 + f_2 \\ &Subject\ to && -8 \le D_1 \le 8 \end{aligned} \tag{17}$$

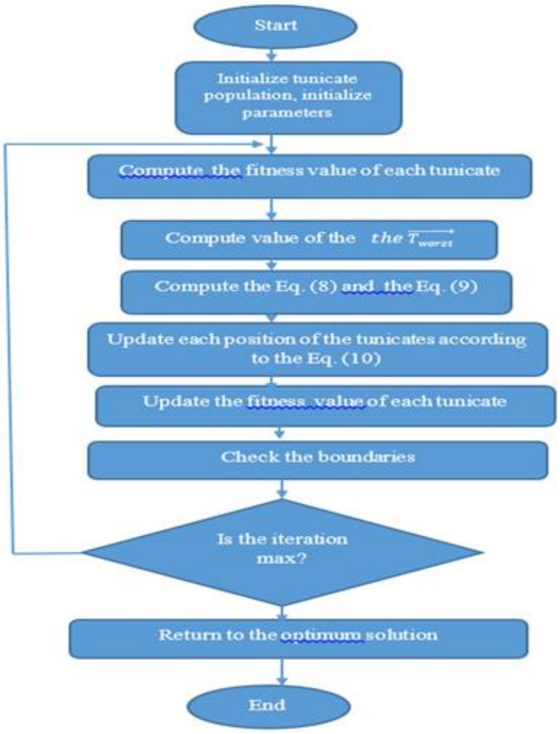

Figure 1 The flowchart of ETSA

3. CONCLUSIONS AND DISCUSSION

All the applications in this section are coded on the Matlab program, and a PC with a Corel i5 processor and 12 GB ram has been chosen for their performance.

3.1. Parameter Analyzes

X_{min} and X_{max} values are taken as 1 and 4, respectively. Necessary parameter analyzes were made by Kaur et al. [10].

In order to show the effect of population size on ETSA, five different values (10, 30, 50, 80, and 100) were chosen randomly. In the tests performed, the maximum iteration value was taken as 500 and the problem size as 30. Thirteen different unimodal and multimodal benchmark test functions have been selected. The details of the benchmark test functions used for the test are given in the appendix. Each test function was run 20 times independently, and the mean (MN) and standard deviation values (SDD) of the obtained results were calculated. The best results are marked in bold. The means and standard deviations of the results are shown in Table 1. Looking at the results, it is seen that the population size has an increasing proportion of the results.

Table 1 The mean (MN) and standard deviation (SDD) results of the ETSA for different population sizes (P=population size)

F	ETSA									
	P=10		P=30		P=50		P=80		P=100	
	MN	SDD	MN	SDD	MN	SDD	MN	SDD	MN	SDD
F1	3.23E-207	**0.0000**	1.48E-233	**0.0000**	2.13E-242	**0.0000**	6.72E-250	**0.0000**	**7.55E-253**	**0.0000**
F2	4.39E-110	**0.0000**	8.60E-124	**0.0000**	1.24E-128	**0.0000**	2.25E-132	**0.0000**	**1.75E-134**	**0.0000**
F3	5.61E-187	**0.0000**	3.74E-205	**0.0000**	1.27E-210	**0.0000**	1.60E-215	**0.0000**	**1.65E-216**	**0.0000**
F4	1.96E-98	**0.0000**	2.03E-107	**0.0000**	4.28E-110	**0.0000**	3.33E-112	**0.0000**	**5.53E-113**	**0.0000**
F5	2.87E+01	**0.2862**	2.86E+01	0.3534	2.87E+01	0.3092	2.87E+01	0.3055	**2.85E+01**	0.3587
F6	5.21E+00	0.6730	4.67E+00	0.6206	4.48E+00	0.6550	**3.95E+00**	**0.5485**	3.98E+00	0.6433
F7	2.85E-04	0.0002	8.17E-05	0.0001	5.72E-05	0.0001	**1.96E-05**	**0.0000**	2.59E-05	**0.0000**
F8	-3.40E+03	458.6395	-3.44E+03	464.1231	-3.41E+03	**312.9168**	**-3.81E+03**	422.5721	-3.35E+03	380.1744
F9	**0.00E+00**	**0.0000**	**0.00E+00**	**0.0000**	**0.00E+00**	**0.0000**	**0.00E+00**	**0.0000**	**0.00E+00**	**0.0000**
F10	**4.44E-15**	**0.0000**	**4.44E-15**	**0.0000**	**4.44E-15**	**0.0000**	**4.44E-15**	**0.0000**	**4.44E-15**	**0.0000**
F11	4.81E-03	0.0119	2.98E-03	0.0081	2.61E-03	0.0069	1.03E-03	0.0032	**8.87E-04**	**0.0027**
F12	9.20E-01	0.2522	6.33E-01	0.2362	5.69E-01	0.1960	5.40E-01	0.1592	**4.90E-01**	**0.1264**
F13	2.87E+00	**0.0792**	2.64E+00	0.1873	2.72E+00	0.1514	2.70E+00	0.1343	**2.61E+00**	0.1867

3.2. Comparison of TSA and ETSA

TSA and ETSA were compared on various unimodal, multi-modal, and fixed-dimensional benchmark test functions in terms of five different comparison criteria (mean, standard deviation, time, best, and worst). The details of the benchmark test functions used for the test are given in the appendix. TSA and ETSA were run 20 times for each benchmark function. The parameter settings used in the comparisons are shown in Table 2. Comparison results are shown in Table 3 and Table 4.

According to the average results, ETSA achieved better results than TSA in all benchmark functions except F8 and F20. The standard deviation results show a parallel success with the average results. According to the time results, TSA works in a shorter time on average. According to Best results, ETSA achieved better results than TSA in 18 of 23 test functions.

Table 2 Parameter settings

Parameter	Value
Population size	30
Dimension	{30, 2, 3, 4, 6}
Maximum iteration	500
X_{min}	1
X_{max}	4

Table 3 The mean (MN), standard deviation (SDD), and Time (T) results of the TSA and ETSA

F	TSA			ETSA		
	MN	**SDD**	**T**	**MN**	**SDD**	**T**
F1	7.84E-19	**0.000000**	**2.87E-01**	**1.48E-233**	**0.000000**	9.21E-01
F2	1.41E-99	5.5E-99	**3.09E-01**	**8.60E-124**	**1.3E-123**	9.68E-01
F3	1.71E-18	**0.000000**	**1.02E+00**	**3.74E-205**	**0.000000**	1.49E+00
F4	5.49E-92	1.56E-91	**3.18E-01**	**2.03E-107**	**5.3E-107**	9.54E-01
F5	2.87E+01	**0.296845**	**3.68E-01**	**2.86E+01**	0.353399	1.15E+00
F6	6.09E+00	0.685039	**3.31E-01**	**4.67E+00**	**0.620591**	1.20E+00
F7	8.38E-05	7.01E-05	**4.56E-01**	**8.17E-05**	**6.94E-05**	1.45E+00
F8	**-3.49E+03**	**440.0958**	**3.73E-01**	-3.44E+03	464.1231	1.24E+00
F9	8.68E+00	24.42137	**3.64E-01**	**0.00E+00**	**0.000000**	1.29E+00
F10	**4.44E-15**	**7.89E-31**	**3.85E-01**	**4.44E-15**	**7.89E-31**	1.00E+00
F11	5.42E-03	0.010736	**3.71E-01**	**2.98E-03**	**0.0081**	1.25E+00
F12	1.10E+00	0.408235	**7.03E-01**	**6.33E-01**	**0.236177**	1.36E+00
F13	2.65E+00	0.235752	**6.93E-01**	**2.64E+00**	**0.187306**	1.42E+00
F14	1.16E+01	4.802528	**1.16E+00**	**7.54E+00**	**4.61054**	2.25E+00
F15	3.77E-03	0.007128	**2.73E-01**	**2.58E-03**	**0.005938**	1.20E+00
F16	-1.02E+00	0.014488	**2.32E-01**	**-1.03E+00**	**0.009485**	9.17E-01
F17	**3.99E-01**	**0.001661**	**2.12E-01**	**3.99E-01**	0.002015	1.03E+00
F18	1.52E+01	24.87551	**2.22E-01**	**7.45E+00**	**19.40461**	8.95E-01
F19	**-3.86E+00**	**0.002301**	**3.43E-01**	**-3.86E+00**	0.002816	1.18E+00
F20	**-3.19E+00**	**0.080977**	**3.50E-01**	-3.07E+00	0.196455	1.18E+00
F21	-7.32E+00	1.349929	**3.92E-01**	**-7.40E+00**	**1.254483**	1.20E+00
F22	-5.25E+00	2.34319	**4.23E-01**	**-5.74E+00**	**1.180463**	1.31E+00
F23	-4.50E+00	2.962034	**5.06E-01**	**-5.97E+00**	**2.21391**	1.41E+00

Table 4 The best and worst results of the TSA and ETSA

F	TSA		ETSA	
	Best	**Worst**	**Best**	**Worst**
F1	1.43E-203	1.45E-192	**4.21E-238**	1.37E-232
F2	1.43E-104	2.55E-98	**1.98E-125**	5.31E-123
F3	4.20E-188	1.42E-182	**6.76E-210**	3.35E-204
F4	1.22E-96	6.44E-91	**2.06E-109**	2.44E-106
F5	**2.81E+01**	2.89E+01	**2.81E+01**	2.90E+01
F6	4.83E+00	7.25E+00	**3.39E+00**	6.01E+00
F7	3.44E-06	2.50E-04	**1.63E-06**	2.47E-04
F8	-4.33E+03	-2.81E+03	**-4.60E+03**	-2.50E+03
F9	**0.000000**	1.05E+02	**0.000000**	0.000000
F10	**4.44E-15**	4.44E-15	**4.44E-15**	4.44E-15
F11	**0.000000**	3.74E-02	**0.000000**	3.35E-02
F12	3.73E-01	1.70E+00	**3.39E-01**	1.34E+00
F13	**1.87E+00**	2.90E+00	2.23E+00	2.90E+00
F14	2.98E+00	1.83E+01	**1.99E+00**	1.27E+01
F15	3.17E-04	2.09E-02	**3.14E-04**	2.04E-02
F16	**-1.03E+00**	-1.00E+00	**-1.03E+00**	-1.00E+00
F17	**3.98E-01**	4.04E-01	**3.98E-01**	4.06E-01
F18	**3.00E+00**	8.41E+01	**3.00E+00**	9.20E+01
F19	**-3.86E+00**	-3.85E+00	**-3.86E+00**	-3.85E+00
F20	**-3.31E+00**	-3.06E+00	-3.30E+00	-2.45E+00
F21	**-9.78E+00**	-4.98E+00	-8.79E+00	-4.89E+00
F22	**-9.04E+00**	-9.11E-01	-7.98E+00	-3.25E+00
F23	**-1.03E+01**	-1.62E+00	-9.97E+00	-1.80E+00

3.3. Comparison of ETSA With Other Algorithms

ETSA and some heuristic algorithms selected from the literature were compared in 23 benchmark functions in terms of four different comparison criteria (mean, standard deviation, time, and best). These algorithms are Jaya Algorithm [11], Skill Optimization Algorithm (SOA) [12], and Slime Mould Algorithm (SMA) [13]. All algorithms were run 20 times for each benchmark function. The parameter settings used in the comparisons are shown in Table 5. Comparison results are shown in Tables 6-9.

According to the results, ETSA achieved better results than Jaya. However, it could not pass SMA and SOA in many functions. ETSA was the most successful algorithm after SMA and SOA. The results show that ETSA can be preferred for continuous optimization problems in the literature.
The Wilcoxon statistical test is a test of significance used to test whether there is a difference between measurements of the same individuals at two different times or situations. Wilcoxon test results for ETSA and other algorithms (Jaya, SOA SMA, and TSA) are shown in Table 10. Convergence plots of the ETSA and other algorithms for various benchmark functions show in Figure 2 and Figure 3.

Table 5 Parameter settings

Parametre	**Değer**
Population size	30
Dimension	{30, 2, 3, 4, 6}
Maximum iteration	500
(X_{min}, X_{max}) for TSA and ETSA	1,4
z for SMA	0.03

Table 6 The comparisons of the mean (MN) results of the ETSA and other algorithms

F	TSA	Jaya	SOA	SMA	ETSA
F1	7.84E-19	3.79E+02	1.28E-219	**6.74E-303**	1.48E-233
F2	1.41E-99	5.70E+00	9.33E-112	**1.82E-152**	8.60E-124
F3	1.71E-18	1.29E+05	1.47E-186	**1.31E-300**	3.74E-205
F4	5.49E-92	6.34E+01	3.40E-107	**7.54E-166**	2.03E-107
F5	2.87E+01	4.61E+05	**0.000000**	3.88E+00	2.86E+01
F6	6.09E+00	2.95E+02	**0.000000**	6.64E-03	4.67E+00
F7	8.38E-05	7.14E+00	**6.37E-05**	1.82E-04	8.17E-05
F8	-3.49E+03	-5.19E+03	-6.49E+03	**-1.26E+04**	-3.44E+03
F9	8.68E+00	1.28E+02	**0.00E+00**	**0.000000**	**0.000000**
F10	4.44E-15	1.88E+01	**8.88E-16**	**8.88E-16**	4.44E-15
F11	5.42E-03	3.78E+00	**0.000000**	**0.000000**	2.98E-03
F12	1.10E+00	9.74E+05	**1.57E-32**	6.50E-03	6.33E-01
F13	2.65E+00	1.24E+06	**1.35E-32**	5.02E-03	2.64E+00
F14	1.16E+01	3.16E+00	2.43E+00	**9.98E-01**	7.54E+00
F15	3.77E-03	7.02E-04	**3.22E-04**	6.53E-04	2.58E-03
F16	-1.02E+00	**-1.03E+00**	**-1.03E+00**	**-1.03E+00**	**-1.03E+00**
F17	3.99E-01	NAN	**3.98E-01**	**3.98E-01**	3.99E-01
F18	1.52E+01	9.79E+00	**3.00E+00**	**3.00E+00**	7.45E+00
F19	**-3.86E+00**	**-3.86E+00**	**-3.86E+00**	**-3.86E+00**	**-3.86E+00**
F20	-3.19E+00	-3.22E+00	**-3.31E+00**	-3.26E+00	-3.07E+00
F21	-7.32E+00	-4.74E+00	-9.64E+00	**-1.02E+01**	-7.40E+00
F22	-5.25E+00	-5.66E+00	-1.01E+01	**-1.04E+01**	-5.74E+00
F23	-4.50E+00	-5.85E+00	**-1.05E+01**	**-1.05E+01**	-5.97E+00
Rank	**5**	**4**	**2**	**1**	**3**

Table 7 The comparisons of the standard deviation (SDD) results of the ETSA and other algorithms

F	TSA	Jaya	SOA	SMA	ETSA
F1	**0.000000**	477.6474	**0.000000**	**0.0000000**	**0.000000**
F2	5.5E-99	6.714119	3.3E-111	**7.9E-152**	1.3E-123
F3	**0.000000**	37118.06	**0.000000**	**0.000000**	**0.000000**
F4	1.56E-91	5.932388	1.2E-106	**0.000000**	5.3E-107
F5	0.296845	644275.3	**0.000000**	8.109989	0.353399
F6	0.685039	202.2773	**0.000000**	0.004869	0.620591
F7	7.01E-05	6.692799	**5.38E-05**	0.000156	6.94E-05
F8	440.0958	2145.906	1033.639	**0.313121**	464.1231
F9	24.42137	42.94716	**0.000000**	**0.000000**	0.000000
F10	7.89E-31	1.349448	**0.000000**	**0.000000**	7.89E-31
F11	0.010736	3.181002	**0.000000**	**0.000000**	0.0081
F12	0.408235	2146562	**5.47E-48**	0.008084	0.236177
F13	0.235752	2289017	**2.74E-48**	0.004642	0.187306
F14	4.802528	3.146973	1.854209	**5.11E-13**	4.61054
F15	0.007128	0.000577	**1.95E-05**	0.00033	0.005938
F16	0.014488	0.003009	**1.89E-10**	7.93E-10	0.009485
F17	0.001661	NAN	**5.19E-09**	3.5E-08	0.002015
F18	24.87551	11.42437	**8.95E-10**	9.47E-09	19.40461
F19	0.002301	**6.65E-15**	1.19E-05	1.83E-07	0.002816
F20	0.080977	0.04338	**0.031431**	0.0593	0.196455
F21	1.349929	2.56062	1.52937	**0.00024**	1.254483
F22	2.34319	3.555628	1.158411	**0.000315**	1.180463
F23	2.962034	3.530381	**8.61E-05**	0.000274	2.21391
Rank	**4**	**5**	**1**	**2**	**3**

Table 8 The comparisons of the time (T) results of the ETSA and other algorithms

F	TSA	Jaya	SOA	SMA	ETSA
F1	2.87E-01	2.04E+00	**2.50E-01**	2.98E+00	9.21E-01
F2	3.09E-01	2.18E+00	**2.40E-01**	3.66E+00	9.68E-01
F3	**1.02E+00**	1.06E+01	1.26E+00	3.43E+00	1.49E+00
F4	**3.18E-01**	2.34E+00	3.48E-01	3.40E+00	9.54E-01
F5	3.68E-01	2.54E+00	**2.72E-01**	3.13E+00	1.15E+00
F6	3.31E-01	1.93E+00	**2.58E-01**	3.13E+00	1.20E+00
F7	**4.56E-01**	3.19E+00	6.83E-01	3.12E+00	1.45E+00
F8	**3.73E-01**	2.62E+00	4.02E-01	3.10E+00	1.24E+00
F9	3.64E-01	2.50E+00	**3.12E-01**	3.36E+00	1.29E+00
F10	**3.85E-01**	2.81E+00	3.93E-01	3.36E+00	1.00E+00
F11	3.71E-01	2.94E+00	**3.40E-01**	3.29E+00	1.25E+00
F12	**7.03E-01**	6.15E+00	9.64E-01	3.55E+00	1.36E+00
F13	**6.93E-01**	6.86E+00	8.14E-01	3.77E+00	1.42E+00
F14	**1.16E+00**	1.31E+01	1.65E+00	1.30E+00	2.25E+00
F15	**2.73E-01**	2.06E+00	4.22E-01	7.92E-01	1.20E+00
F16	**2.32E-01**	2.07E+00	3.38E-01	6.10E-01	9.17E-01
F17	**2.12E-01**	NAN	2.18E-01	5.92E-01	1.03E+00
F18	**2.22E-01**	1.68E+00	3.16E-01	5.76E-01	8.95E-01
F19	**3.43E-01**	2.77E+00	4.72E-01	8.16E-01	1.18E+00
F20	**3.50E-01**	2.77E+00	5.64E-01	1.01E+00	1.18E+00
F21	**3.92E-01**	6.49E+00	5.70E-01	1.21E+00	1.20E+00
F22	**4.23E-01**	7.56E+00	6.34E-01	8.63E-01	1.31E+00
F23	**5.06E-01**	8.23E+00	7.44E-01	1.02E+00	1.41E+00
Rank	**1**	**5**	**2**	**4**	**3**

Table 9 The comparisons of the best results of the ETSA and other algorithms

F	TSA	Jaya	SOA	SMA	ETSA
F1	1.43E-203	9.27E+00	1.25E-242	**0.00E+00**	4.21E-238
F2	1.43E-104	7.23E-01	1.35E-121	**3.95E-267**	1.98E-125
F3	4.20E-188	7.04E+04	1.55E-210	**0.00E+00**	6.76E-210
F4	1.22E-96	4.51E+01	2.46E-118	**3.52E-293**	2.06E-109
F5	2.81E+01	8.42E+03	**0.000000**	3.90E-03	2.81E+01
F6	4.83E+00	5.15E+01	**0.000000**	1.50E-03	3.39E+00
F7	3.44E-06	5.63E-01	**6.14E-07**	3.77E-05	1.63E-06
F8	-4.33E+03	-8.21E+03	-8.66E+03	**-1.26E+04**	-4.60E+03
F9	**0.000000**	6.68E+01	**0.00E+00**	**0.000000**	**0.000000**
F10	4.44E-15	1.51E+01	**8.88E-16**	**8.88E-16**	4.44E-15
F11	**0.000000**	1.17E+00	**0.00E+00**	**0.000000**	**0.000000**
F12	3.73E-01	4.89E+01	**1.57E-32**	3.26E-05	3.39E-01
F13	1.87E+00	7.25E+00	**1.35E-32**	6.31E-05	2.23E+00
F14	2.98E+00	**9.98E-01**	**9.98E-01**	**9.98E-01**	1.99E+00
F15	3.17E-04	3.08E-04	**3.07E-04**	3.08E-04	3.14E-04
F16	**-1.03E+00**	**-1.03E+00**	**-1.03E+00**	**-1.03E+00**	**-1.03E+00**
F17	**3.98E-01**	NAN	**3.98E-01**	**3.98E-01**	**3.98E-01**
F18	**3.00E+00**	**3.00E+00**	**3.00E+00**	**3.00E+00**	**3.00E+00**
F19	**-3.86E+00**	**-3.86E+00**	**-3.86E+00**	**-3.86E+00**	**-3.86E+00**
F20	-3.31E+00	**-3.32E+00**	**-3.32E+00**	**-3.32E+00**	-3.30E+00
F21	-9.78E+00	**-1.02E+01**	**-1.02E+01**	**-1.02E+01**	-8.79E+00
F22	-9.04E+00	**-1.04E+01**	**-1.04E+01**	**-1.04E+01**	-7.98E+00
F23	-1.03E+01	**-1.05E+01**	**-1.05E+01**	**-1.05E+01**	-9.97E+00
Rank	**5**	**3**	**1**	**2**	**4**

Table 10 The Wilcoxon test results of the ETSA and other algorithms

F	ETSA-Jaya		ETSA-SOA		ETSA-SMA		ETSA-TSA	
	p-value	**h-value**	**p-value**	**h-value**	**p-value**	**h-value**	**p-value**	**h-value**
F1	0.000248	1	2.92E-05	1	0.00047	1	0.000248	1
F2	0.000248	1	0.000248	1	0.000248	1	0.000248	1
F3	0.000248	1	6.67E-06	1	0.000365	1	0.000248	1
F4	0.000228	1	0.000758	1	6.80E-08	1	0.000248	1
F5	0.000218	1	7.72E-09	1	1.61E-07	1	0.6747	0
F6	0.000248	1	0.000212	1	0.000248	1	0.000475	1
F7	0.000248	1	0.350702	1	0.003639	1	0.7353	0
F8	0.1264	0	7.90E-08	1	6.80E-08	1	0.7764	0
F9	8.01E-09	1	NaN	0	NaN	0	3.31E-06	1
F10	7.68E-09	1	4.68E-10	1	4.68E-10	1	NAN	0
F11	1.96E-08	1	0.080631	0	0.080631	0	0.4162	0
F12	6.80E-08	1	8.01E-09	1	6.80E-08	1	7.58E-04	1
F13	6.47E-08	1	7.51E-09	1	6.47E-08	1	0.8815	0
F14	7.98E-04	1	8.59E-06	1	6.89E-09	1	0.0121	1
F15	0.8392	0	1.81E-05	1	1	0	0.7557	0
F16	0.0499	1	8.01E-09	1	1.13E-08	1	0.2616	0
F17	NAN	0	3.46E-08	1	6.76E-08	1	0.7150	0
F18	1.21E-06	1	1.13E-08	1	1.51E-08	1	0.0018	1
F19	8.01E-09	1	6.80E-08	1	6.79E-08	1	0.1333	0
F20	4.16E-04	1	1.43E-07	1	2.60E-05	1	0.0114	1
F21	6.15E-04	1	9.56E-06	1	6.61E-08	1	0.5975	0
F22	0.3648	0	2.56E-07	1	6.80E-08	1	0.6359	0
F23	0.6750	0	6.80E-08	1	6.80E-08	1	0.0601	0

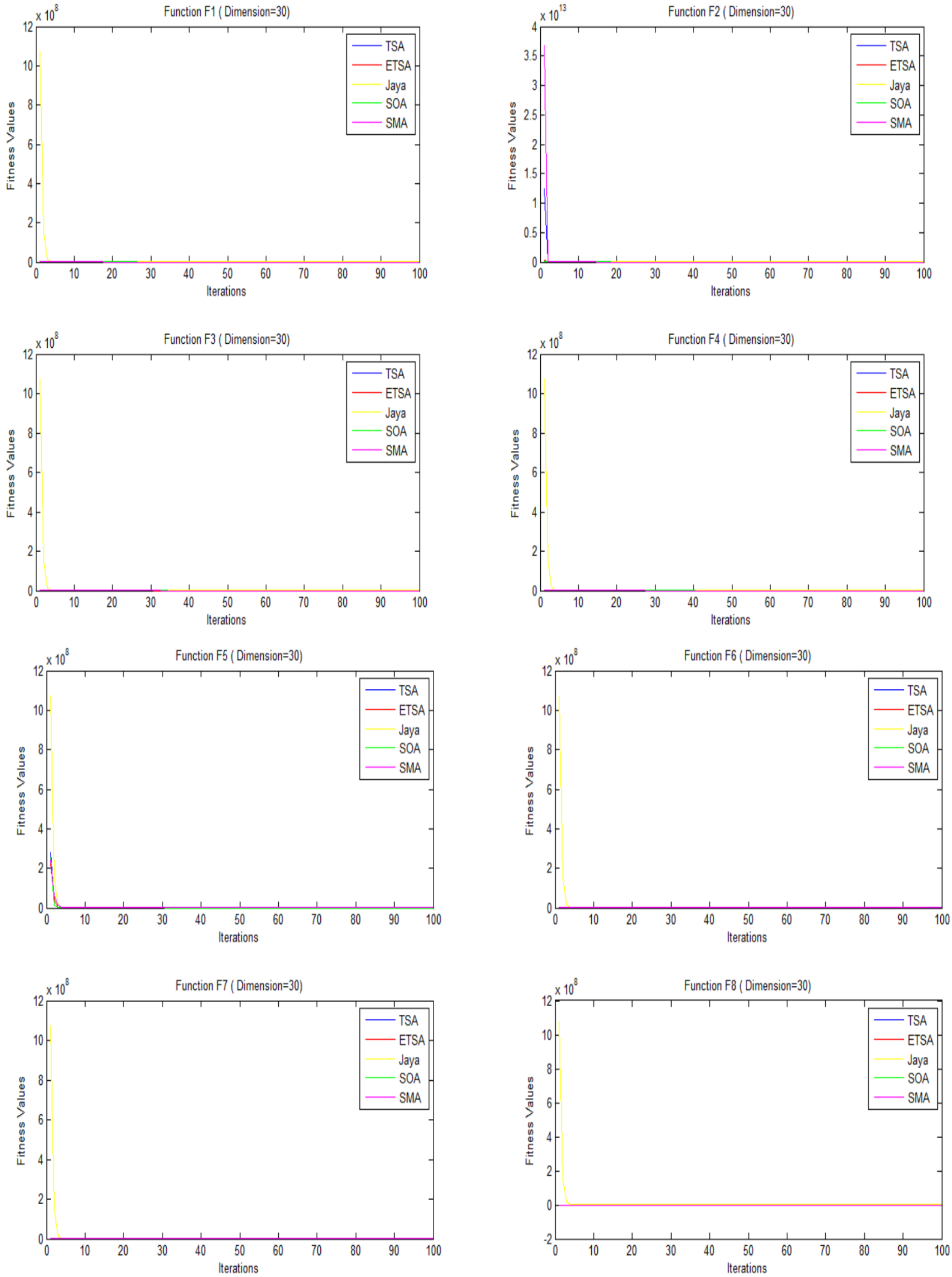

Figure 2 Convergence plots of the ETSA and other algorithms for various benchmark functions (F1, F2, F3, F4, F5, F6, F7, and F8) in dimension=30

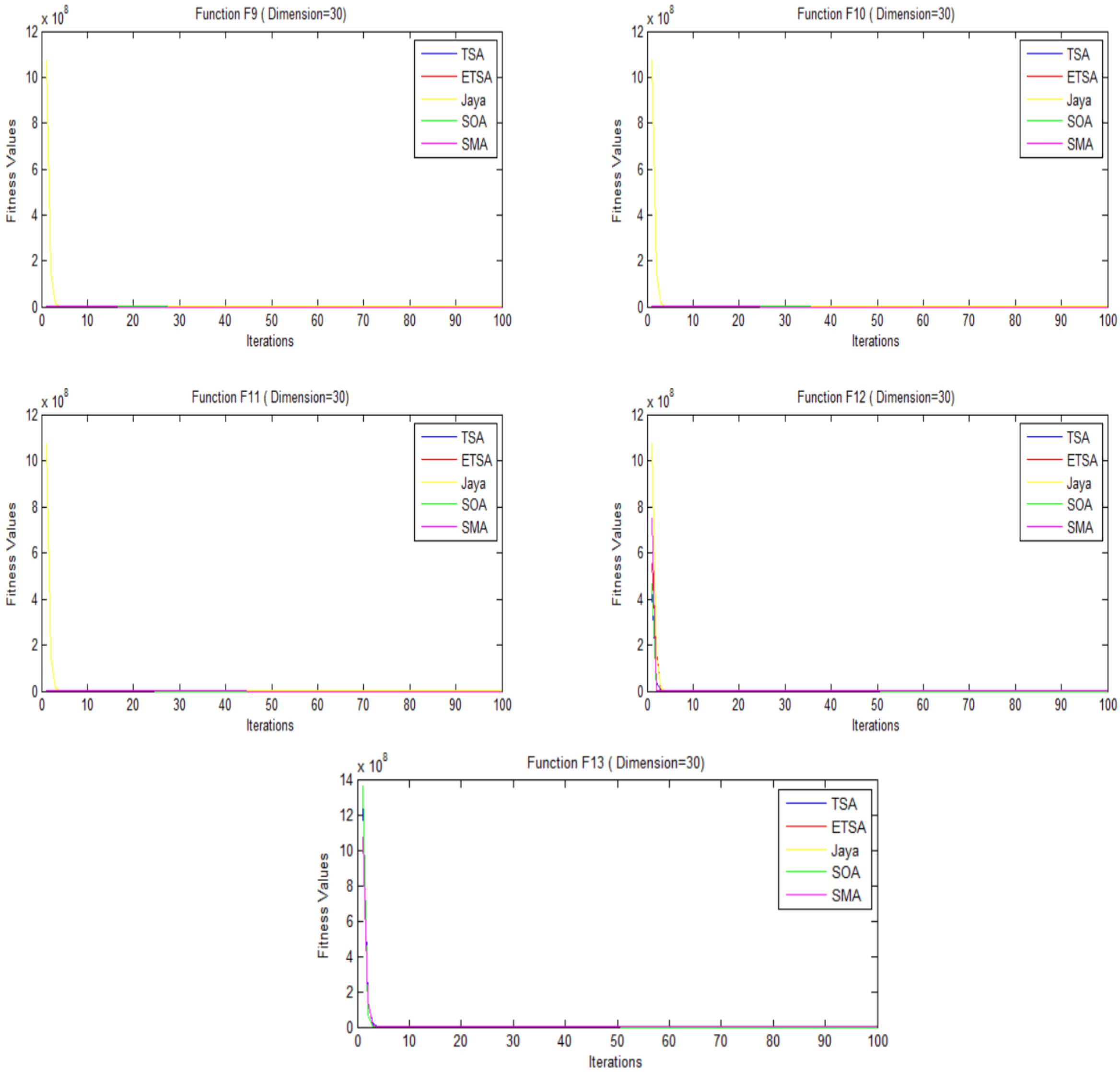

Figure 3 Convergence plots of the ETSA and other algorithms for various benchmark functions (F9, F10, F11, F12, and F13) in dimension=30

3.4. ETSA For Big Data Optimization

The success of TSA and ETSA was tested in detail on six different big data sets. During these tests, five different population sizes (10, 20, 30, 50, 100) were determined with three different maximum iteration values (300, 500, 1000). TSA and ETSA algorithms were run in 20 independent studies on each big data set. Obtained results were compared according to four different criteria (best, mean (MN), standard deviation (SDD), and time (T)). In addition, the results were compared with some heuristic algorithms selected from the literature and proposed in recent years. These algorithms are Jaya Algorithm [11], Skill Optimization Algorithm (SOA) [12], and Slime Mould

Algorithm (SMA) [13]. The parameter settings used in the comparisons are shown in Table 11.

Comparison results of TSA and ETSA for maximum iteration={300, 500, 1000} are shown in Tables 12-17. Comparison results of ETSA and other algorithms are shown in Table 18. Wilcoxon test results for ETSA and other algorithms (TSA, Jaya, SOA, and SMA) are shown in Table 19.

According to the results, ETSA achieved better results than TSA. Thus, it has been proven that In terms of working time, TSA worked in a shorter time than ETSA. The reason for this is the codes added for the development of ETSA. It is not a significant time difference. ETSA has been developed. When compared with various algorithms selected from the literature, it achieved the best results after SOA. The results obtained were subjected to the Wilcoxon sign test, which is a statistical test, and it was shown that there was a semantic difference between the results obtained.

Table 11 Parameter settings

Parameters	**Values**
Population size (P)	10, 20, 30, 50, 100
Dimension	1024, 3072, 4868
Maximum iteration	300, 500, 1000
(X_{min}, X_{max}) for TSA and ETSA	1,4
z for SMA	0.03

Table 12 The results of the TSA in big data datasets for maximum iteration=300 (P=population size)

Datasets	**P=10**	**P=20**	**P=30**	**P=50**	**P=100**
EE4					
Best	1.41E+00	1.36E+00	1.37E+00	1.35E+00	**1.33E+00**
MN	1.47E+00	1.43E+00	1.43E+00	1.41E+00	**1.39E+00**
SDD	3.40E-02	3.86E-02	3.02E-02	3.27E-02	**2.59E-02**
T	0.9165	1.6946	3.5140	4.8225	7.3009
EE4N					
Best	1.42E+00	1.40E+00	1.39E+00	1.35E+00	**1.32E+00**
MN	1.48E+00	1.46E+00	1.42E+00	1.41E+00	**1.38E+00**
SDD	3.71E-02	3.16E-02	3.04E-02	2.95E-02	**2.87E-02**
T	0.9149	1.6457	2.3803	3.7127	7.2040
EE12					
Best	1.56E+00	1.56E+00	1.54E+00	1.52E+00	**1.48E+00**
MN	1.63E+00	1.59E+00	1.58E+00	1.56E+00	**1.53E+00**
SDD	2.87E-02	**1.66E-02**	2.75E-02	2.03E-02	2.32E-02
T	2.0750	5.8661	5.7513	9.4226	20.1355
EE12N					
Best	1.57E+00	1.56E+00	1.54E+00	1.52E+00	**1.51E+00**
MN	1.63E+00	1.60E+00	1.59E+00	1.56E+00	**1.54E+00**
SDD	2.79E-02	2.57E-02	2.67E-02	2.11E-02	**1.94E-02**
T	2.0703	5.5020	5.8107	11.8639	35.1981
EE19					
Best	1.60E+00	1.58E+00	1.57E+00	1.53E+00	**1.51E+00**
MN	1.66E+00	1.63E+00	1.61E+00	1.58E+00	**1.57E+00**
SDD	2.68E-02	**2.09E-02**	2.58E-02	2.99E-02	3.12E-02
T	6.0403	11.4253	17.6720	33.0520	53.9655
EE19N					
Best	1.60E+00	1.59E+00	1.57E+00	**1.54E+00**	**1.54E+00**
MN	1.66E+00	1.63E+00	1.61E+00	1.60E+00	**1.57E+00**
SDD	2.74E-02	2.25E-02	3.20E-02	2.48E-02	**2.09E-02**
T	4.6871	12.8426	19.2210	33.0938	64.8880

Table 13 The results of the TSA in big data datasets for maximum iteration=500 (P=population size)

Datasets		P=10	P=20	P=30	P=50	P=100
EE4						
	Best	1.41E+00	1.37E+00	1.36E+00	1.35E+00	**1.32E+00**
	MN	1.46E+00	1.43E+00	1.42E+00	1.40E+00	**1.36E+00**
	SDD	2.80E-02	3.73E-02	2.94E-02	2.82E-02	**1.97E-02**
	T	2.2915	4.2904	6.2368	9.4524	18.6667
EE4N						
	Best	1.37E+00	1.36E+00	1.36E+00	1.36E+00	**1.35E+00**
	MN	1.45E+00	1.43E+00	1.42E+00	1.40E+00	**1.38E+00**
	SDD	3.61E-02	3.79E-02	2.85E-02	**2.56E-02**	2.61E-02
	T	1.5895	4.1892	6.4027	10.1032	22.2316
EE12						
	Best	1.56E+00	1.53E+00	1.52E+00	1.51E+00	**1.49E+00**
	MN	1.62E+00	1.58E+00	1.57E+00	1.54E+00	**1.53E+00**
	SDD	2.72E-02	2.63E-02	**1.93E-02**	1.98E-02	2.10E-02
	T	5.6007	11.1384	14.5790	24.7584	55.0372
EE12N						
	Best	1.55E+00	1.53E+00	1.50E+00	1.50E+00	**1.49E+00**
	MN	1.61E+00	1.58E+00	1.57E+00	1.55E+00	**1.53E+00**
	SDD	2.32E-02	**2.06E-02**	2.26E-02	2.29E-02	2.23E-02
	T	3.5387	10.5012	14.5954	25.9696	52.1192
EE19						
	Best	1.61E+00	1.54E+00	1.57E+00	1.56E+00	**1.52E+00**
	MN	1.65E+00	1.62E+00	1.60E+00	1.58E+00	**1.57E+00**
	SDD	2.70E-02	2.52E-02	2.04E-02	**1.91E-02**	2.76E-02
	T	11.4051	21.0422	31.0650	38.1544	85.7928
EE19N						
	Best	1.61E+00	1.57E+00	1.58E+00	**1.54E+00**	**1.54E+00**
	MN	1.66E+00	1.62E+00	1.61E+00	1.58E+00	**1.57E+00**
	SDD	2.38E-02	2.20E-02	2.19E-02	2.64E-02	**1.97E-02**
	T	13.4974	14.2832	38.5597	106.2589	151.6567

Table 14 The results of the TSA in big data datasets for maximum iteration=1000 (P=population size)

Datasets		P=10	P=20	P=30	P=50	P=100
EE4						
	Best	1.37E+00	1.36E+00	1.38E+00	1.35E+00	**1.31E+00**
	MN	1.43E+00	1.41E+00	1.43E+00	1.39E+00	**1.36E+00**
	SDD	3.27E-02	2.75E-02	3.70E-02	2.66E-02	**2.54E-02**
	T	6.8589	12.2524	13.4907	29.1908	39.8686
EE4N						
	Best	1.39E+00	1.37E+00	1.35E+00	**1.32E+00**	1.34E+00
	MN	1.44E+00	1.42E+00	1.42E+00	1.38E+00	**1.37E+00**
	SDD	3.91E-02	3.19E-02	4.17E-02	**2.81E-02**	2.83E-02
	T	6.4894	12.1303	19.6702	20.3234	24.5076
EE12						
	Best	1.55E+00	1.53E+00	1.50E+00	1.49E+00	**1.48E+00**
	MN	1.59E+00	1.57E+00	1.58E+00	1.54E+00	**1.52E+00**
	SDD	2.46E-02	2.11E-02	3.74E-02	2.10E-02	**2.01E-02**
	T	12.7430	31.1192	56.5263	34.6869	66.9431

Table 14 The results of the TSA in big data datasets for maximum iteration=1000 (P=population size) (Continue)

Datasets		P=10	P=20	P=30	P=50	P=100
EE12N						
	Best	1.56E+00	1.53E+00	1.54E+00	1.49E+00	**1.47E+00**
	MN	1.60E+00	1.56E+00	1.58E+00	1.54E+00	**1.52E+00**
	SDD	2.43E-02	**1.99E-02**	2.24E-02	2.01E-02	2.48E-02
	T	7.3831	14.0870	35.5943	33.4560	67.7289
EE19						
	Best	1.59E+00	1.57E+00	1.58E+00	1.52E+00	**1.50E+00**
	MN	1.63E+00	1.59E+00	1.62E+00	1.56E+00	**1.56E+00**
	SDD	2.63E-02	2.28E-02	3.17E-02	2.59E-02	**2.21E-02**
	T	15.6217	24.2274	65.7384	59.4652	118.7629
EE19N						
	Best	1.59E+00	1.57E+00	1.55E+00	1.53E+00	**1.51E+00**
	MN	1.63E+00	1.61E+00	1.60E+00	1.57E+00	**1.55E+00**
	SDD	2.17E-02	2.15E-02	2.74E-02	**2.00E-02**	2.64E-02
	T	21.6556	44.0970	100.3609	102.0027	216.4222

Table 15 The results of the ETSA in big data datasets for maximum iteration=300 (P=population size)

Datasets		P=10	P=20	P=30	P=50	P=100
EE4						
	Best	1.42E+00	1.41E+00	1.39E+00	1.38E+00	**1.37E+00**
	MN	1.50E+00	1.46E+00	1.44E+00	1.43E+00	**1.41E+00**
	SDD	3.92E-02	3.18E-02	3.23E-02	3.45E-02	**2.96E-02**
	T	1.4530	4.0206	5.4442	15.5729	31.3489
EE4N						
	Best	1.45E+00	1.41E+00	1.38E+00	1.37E+00	**1.35E+00**
	MN	1.52E+00	1.47E+00	1.45E+00	1.43E+00	**1.40E+00**
	SDD	4.17E-02	3.35E-02	3.39E-02	**2.78E-02**	2.92E-02
	T	1.7041	3.9030	5.4782	7.5168	23.8819
EE12						
	Best	1.61E+00	1.57E+00	1.52E+00	1.54E+00	**1.53E+00**
	MN	1.66E+00	1.62E+00	1.59E+00	1.59E+00	**1.57E+00**
	SDD	2.33E-02	2.62E-02	2.94E-02	3.02E-02	**1.95E-02**
	T	3.7527	10.4922	17.7711	34.0430	84.6269
EE12N						
	Best	1.59E+00	1.56E+00	1.55E+00	1.53E+00	**1.51E+00**
	MN	1.65E+00	1.62E+00	1.60E+00	1.58E+00	**1.56E+00**
	SDD	2.94E-02	2.85E-02	2.52E-02	**2.49E-02**	3.54E-02
	T	4.8291	8.7621	16.4479	35.4610	95.6271
EE19						
	Best	1.61E+00	1.59E+00	1.60E+00	1.55E+00	**1.52E+00**
	MN	1.67E+00	1.65E+00	1.64E+00	1.61E+00	**1.60E+00**
	SDD	3.18E-02	**2.90E-02**	2.96E-02	3.21E-02	3.39E-02
	T	8.2209	16.9738	34.4714	70.4807	190.8606
EE19N						
	Best	1.61E+00	1.57E+00	1.58E+00	1.57E+00	**1.52E+00**
	MN	1.68E+00	1.64E+00	1.64E+00	1.62E+00	**1.59E+00**
	SDD	4.04E-02	3.58E-02	**2.86E-02**	3.36E-02	3.36E-02
	T	8.8840	18.5949	32.2787	75.0291	148.0824

Table 16 The results of the ETSA in big data datasets for maximum iteration=500 (P=population size)

Datasets		P=10	P=20	P=30	P=50	P=100
EE4						
	Best	1.42E+00	1.42E+00	1.39E+00	1.37E+00	**1.35E+00**
	MN	1.50E+00	1.47E+00	1.44E+00	1.41E+00	**1.39E+00**
	SDD	4.24E-02	3.05E-02	2.75E-02	**2.28E-02**	2.56E-02
	T	2.7479	3.7079	5.5867	12.0395	27.0730
EE4N						
	Best	1.42E+00	1.41E+00	1.36E+00	1.36E+00	**1.35E+00**
	MN	1.48E+00	1.46E+00	1.43E+00	1.42E+00	**1.40E+00**
	SDD	3.33E-02	2.92E-02	3.10E-02	2.73E-02	**2.71E-02**
	T	1.9530	3.7568	5.5512	12.1127	29.3235
EE12						
	Best	1.56E+00	1.56E+00	1.54E+00	1.53E+00	**1.50E+00**
	MN	1.64E+00	1.61E+00	1.59E+00	1.57E+00	**1.56E+00**
	SDD	3.16E-02	**2.34E-02**	3.08E-02	2.70E-02	2.82E-02
	T	11.2345	25.0109	35.9231	39.8899	142.1579
EE12N						
	Best	1.56E+00	1.55E+00	1.56E+00	1.53E+00	**1.51E+00**
	MN	1.63E+00	1.60E+00	1.60E+00	1.58E+00	**1.55E+00**
	SDD	3.53E-02	3.15E-02	**1.68E-02**	2.71E-02	2.22E-02
	T	5.5528	11.9986	19.6726	41.4899	110.5248
EE19						
	Best	1.59E+00	1.59E+00	1.57E+00	1.56E+00	**1.53E+00**
	MN	1.65E+00	1.65E+00	1.63E+00	1.61E+00	**1.58E+00**
	SDD	4.06E-02	3.20E-02	3.29E-02	**2.88E-02**	3.23E-02
	T	13.1811	28.8617	51.5975	93.6329	330.7623
EE19N						
	Best	1.60E+00	1.57E+00	1.56E+00	1.56E+00	**1.54E+00**
	MN	1.66E+00	1.63E+00	1.62E+00	1.61E+00	**1.59E+00**
	SDD	3.27E-02	3.08E-02	**2.86E-02**	3.30E-02	3.38E-02
	T	8.9610	18.9731	37.0469	70.5432	196.8537

Table 17 The results of the ETSA in big data datasets for maximum iteration=1000 (P=population size)

Datasets		P=10	P=20	P=30	P=50	P=100
EE4						
	Best	1.40E+00	1.35E+00	1.37E+00	1.36E+00	**1.35E+00**
	MN	1.46E+00	1.42E+00	1.41E+00	1.42E+00	**1.40E+00**
	SDD	3.42E-02	3.69E-02	**2.62E-02**	2.64E-02	3.08E-02
	T	4.3305	7.9714	18.1214	23.2406	56.0005
EE4N						
	Best	1.42E+00	1.42E+00	1.36E+00	1.36E+00	**1.34E+00**
	MN	1.47E+00	1.45E+00	1.41E+00	1.41E+00	**1.38E+00**
	SDD	2.90E-02	2.39E-02	**2.98E-02**	3.09E-02	2.31E-02
	T	4.2536	7.9278	11.6828	34.8273	53.0632
EE12						
	Best	1.59E+00	1.55E+00	1.50E+00	1.51E+00	**1.49E+00**
	MN	1.62E+00	1.60E+00	**1.54E+00**	1.56E+00	**1.54E+00**
	SDD	**1.80E-02**	2.51E-02	2.14E-02	2.56E-02	2.21E-02
	T	9.8815	21.1255	46.0445	73.9346	568.3986

Table 17 The results of the ETSA in big data datasets for maximum iteration=1000 (P=population size) (Continue)

Datasets		P=10	P=20	P=30	P=50	P=100
EE12N						
	Best	1.55E+00	1.55E+00	1.51E+00	1.51E+00	**1.48E+00**
	MN	1.62E+00	1.60E+00	1.55E+00	1.57E+00	**1.54E+00**
	SDD	3.36E-02	3.14E-02	**2.30E-02**	2.60E-02	3.07E-02
	T	9.1114	36.2446	29.3428	71.2533	2685.8105
EE19						
	Best	1.58E+00	1.56E+00	1.56E+00	**1.54E+00**	**1.54E+00**
	MN	1.65E+00	1.62E+00	1.59E+00	1.59E+00	**1.58E+00**
	SDD	3.87E-02	**2.83E-02**	1.84E-02	3.11E-02	3.05E-02
	T	18.0155	47.6943	55.7210	186.7582	790.6452
EE19N						
	Best	1.57E+00	1.58E+00	1.56E+00	**1.52E+00**	1.53E+00
	MN	1.65E+00	1.63E+00	1.59E+00	1.59E+00	**1.57E+00**
	SDD	3.64E-02	2.41E-02	2.29E-02	2.74E-02	**2.28E-02**
	T	20.5374	36.7440	59.0029	192.0792	373.6314

Table 18 The comparisons results of the ETSA and other algorithms in big data datasets for maximum iteration=1000 and population size=30

Datasets		Jaya	SOA	SMA	TSA	ETSA
EE4						
	Best	1.17E+01	**1.33E+00**	1.87E+00	1.38E+00	1.37E+00
	MN	1.29E+01	**1.36E+00**	1.92E+00	1.43E+00	1.41E+00
	SDD	6.35E-01	2.38E-02	**2.26E-02**	3.70E-02	2.62E-02
	T	4.00E+01	**4.58E+00**	1.24E+02	13.4907	18.1214
EE4N						
	Best	1.19E+01	**1.27E+00**	1.85E+00	1.35E+00	1.36E+00
	MN	1.30E+01	**1.31E+00**	1.90E+00	1.42E+00	1.41E+00
	SDD	6.49E-01	**2.83E-02**	3.11E-02	4.17E-02	2.98E-02
	T	3.04E+01	**3.94E+00**	1.59E+02	19.6702	11.6828
EE12						
	Best	1.30E+01	**1.43E+00**	1.93E+00	1.50E+00	1.50E+00
	MN	1.39E+01	**1.53E+00**	1.95E+00	1.58E+00	1.54E+00
	SDD	3.37E-01	3.47E-02	**1.39E-02**	3.74E-02	2.14E-02
	T	7.06E+01	**1.13E+01**	4.86E+02	56.5263	46.0445
EE12N						
	Best	1.34E+01	**1.50E+00**	1.93E+00	1.54E+00	1.51E+00
	MN	1.38E+01	**1.55E+00**	1.95E+00	1.58E+00	**1.55E+00**
	SDD	2.44E-01	4.28E-02	**1.41E-02**	2.24E-02	2.30E-02
	T	6.05E+01	**1.14E+01**	3.07E+02	35.5943	29.3428
EE19						
	Best	1.36E+01	**1.53E+00**	1.93E+00	1.58E+00	1.56E+00
	MN	1.40E+01	**1.58E+00**	1.96E+00	1.62E+00	1.59E+00
	SDD	2.14E-01	2.39E-02	1.64E-02	3.17E-02	**1.84E-02**
	T	1.01E+02	**1.57E+01**	2.22E+02	65.7384	55.7210
EE19N						
	Best	1.36E+01	1.56E+00	1.92E+00	**1.55E+00**	1.56E+00
	MN	1.41E+01	1.60E+00	1.96E+00	1.60E+00	**1.59E+00**
	SDD	2.87E-01	2.35E-02	**1.84E-02**	2.74E-02	2.29E-02
	T	1.02E+02	**1.92E+01**	2.37E+02	100.3609	59.0029
Rank		**5**	**1**	**4**	**3**	**2**

Table 19 The Wilcoxon test results of the ETSA and other algorithms

Datasets	ETSA-TSA		ETSA-Jaya		ETSA-SOA		ETSA-SMA	
	p-value	h-value	p-value	h-value	p-value	h-value	p-value	h-value
D4	0.1478	0	6.80E-06	1	9.13E-07	1	6.72E-08	1
D4N	0.2853	0	6.80E-06	1	1.66E-07	1	6.70E-08	1
D12	0.0036	1	6.80E-06	1	8.36E-04	1	6.46E-08	1
D12N	1.16E-04	1	6.80E-06	1	0.0043	1	6.61E-08	1
D19	0.0114	1	6.80E-06	1	8.36E-04	1	6.70E-08	1
D19N	0.3942	0	6.80E-06	1	0.9676	0	6.70E-08	1

4. CONCLUSION

Tunicate Swarm Algorithm (TSA) is proposed by Kaur et al. in 2020. Tunicate Swarm Algorithm (TSA) was inspired by sea creatures called Tunicates. Tunicates are marine creatures capable of locating food sources in the seas. Tunicates adopted two fundamental movements in exploration and exploitation behavior: jet propulsion and herd intelligence. Thanks to these two movements. it performs exploration and exploitation operations. In this study. herd behavior has been updated and Enhanced TSA (ETSA) has been proposed to increase the success of TSA. Two sets of two problems were used to test the success of TSA and ETSA. In the first problem set. twenty-three unimodal. multimodal. and fixed-dimensional multi-modal test functions were selected from the literature. Secondly. the big data optimization problem involving EEG signals is chosen. The results were tested according to the best. mean. standard deviation. and time comparison criteria for TSA and ETSA. and convergence graphs were shown. The performances of TSA and ETSA were tested in detail for five different population sizes (10. 20. 30. 50. and 100) and three different maximum iterations (300. 500. and 1000) on big datasets containing six different EEG signals. TSA and ETSA results were also compared with Jaya. SOA. and SMA algorithms. The results showed that ETSA achieved better results than TSA and showed preferable success for continuous optimization problems.

In further studies. it is expected that the binary form of TSA will be obtained and the feature selection problem will be solved and its success will be demonstrated.

Funding
The author (s) has not received any financial support for the research. authorship. or publication of this study.

Authors' Contribution
The authors contributed equally to the study.

The Declaration of Conflict of Interest/ Common Interest
No conflict of interest or common interest has been declared by the authors.

The Declaration of Ethics Committee Approval
This study does not require ethics committee permission or any special permission.

The Declaration of Research and Publication Ethics
The authors of the paper declare that they comply with the scientific. ethical. and quotation rules of SAUJS in all processes of the paper and that they do not make any falsification of the data collected. In addition. they declare that Sakarya University Journal of Science and its editorial board have no responsibility for any ethical violations that may be encountered

and that this study has not been evaluated in any academic publication environment other than Sakarya University Journal of Science.

REFERENCES

[1] S. Aslan, D. Karaboga, “A genetic Artificial Bee Colony algorithm for signal reconstruction based big data optimization,” Applied Soft Computing, vol. 88, pp. 106053, 2020.

[2] V. N. Gudivada, R. Baeza-Yates, V. V. Raghavan, “Big data: Promises and problems,” Computer, vol. 48, no. 3, pp. 20–23, 2015.

[3] C. W. Tsai, C. -F. Lai, H. -C. Chao, A. V. Vasilakos, “Big data analytics: a survey,” Journal Big Data, vol. 2, no. 1, pp. 21, 2015.

[4] M. Hilbert, “Big data for development: A review of promises and challenges,” Development Policy Review, vol. 34, no. 1, pp. 135–174, 2016.

[5] S. K. Goh, K. C. Tan, A. Al-Mamun, H.A. Abbass, “Evolutionary big optimization (BigOpt) of signals,” In: 2015 IEEE congress on evolutionary computation (CEC), IEEE, 2015, pp. 3332–3339.

[6] Y. Zhang, J. Liu, M. Zhou, Z. Jiang, “A multi-objective memetic algorithm based on decomposition for big optimization problems,” Memetic Computing, vol. 8, no. 1, pp. 45–61, 2016.

[7] S. Elsayed, R. Sarker, “An adaptive configuration of differential evolution algorithms for big data,” in: IEEE Congress on Evolutionary Computation (CEC), IEEE, 2015, pp. 695–702.

[8] M. A. El Majdouli, I. Rbouh, S. Bougrine, B. El Benani, A. A. El Imrani, “Fireworks algorithm framework for Big Data optimization,” Memetic Computing, vol. 8, no. 4, pp. 333–347, 2016.

[9] Z. Cao, L. Wang, X. Hei, Q. Jiang, X. Lu, X. Wang, “A phase based optimization algorithm for big optimization problems,” in: 2016 IEEE Congress on Evolutionary Computation (CEC), IEEE, 2016, pp. 5209–5214.

[10] S. Kaur, L. K. Awasthi, A.L. Sangal, G. Dhiman, “Tunicate Swarm Algorithm: A new bio-inspired based metaheuristic paradigm for global optimization,” Engineering Applications of Artificial Intelligence, vol. 90, pp. 103541, 2020.

[11] R. Rao, “Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems,” International Journal of Industrial Engineering Computations, vol. 7, no. 1, pp. 19-34, 2016.

[12] H. Givi, M. Hubalovska, "Skill optimization algorithm: a new human-based metaheuristic technique,” Computers Materials & Continua, vol. 74, no.1, pp. 179–202, 2023.

[13] S. Li, H. Chen, M. Wang, A. A. Heidari, S. Mirjalili, “Slime mould algorithm: A new method for stochastic optimization,” Future Generation Computer Systems, vol. 111, pp. 300–323, 2020.

[14] S. K. Goh, H.A. Abbass, K. C. Tan, A. Al Mamun, “Artifact removal from EEG using a multi-objective independent component analysis model,” In: International conference on neural information processing, Springer, 2014, pp. 570–577.

[15] H. A. Abbass, "Calibrating independent component analysis with Laplacian reference for real-time EEG artifact removal," In: International conference on neural information processing, Springer, 2014, pp. 68–75.

Appendix: Benchmark functions

A. Unimodal benchmark functions

F1:

$$F_1(x) = \sum_{i=1}^{d} x_i^2 \qquad -100 \le x_i \le 100 \quad f_{min} = 0 \quad d = dimension$$

F2:

$$F_2(x) = \sum_{i=1}^{d} |x_i^2| + \prod_{i=1}^{d} |x_i^2| \qquad -10 \le x_i \le 10 \quad f_{min} = 0 \quad d = dimension$$

F3:

$$F_3(x) = \sum_{i=1}^{d} \left(\sum_{j=1}^{i} x_j \right)^2 \qquad -100 \le x_i \le 100 \quad f_{min} = 0 \quad d = dimension$$

F4:

$$F_4(x) = \max_i \{|x_i|.1 \le i \le d\} \qquad -100 \le x_i \le 100 \quad f_{min} = 0 \quad d = dimension$$

F5:

$$F_5(x) = \sum_{i=1}^{d-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2] \qquad -30 \le x_i \le 30 \quad f_{min} = 0 \quad d = dimension$$

F6:

$$F_6(x) = \sum_{i=1}^{d} ([x_i + 0.5])^2 \qquad -100 \le x_i \le 100 \quad f_{min} = 0 \quad d = dimension$$

F7:

$$F_7(x) = \sum_{i=1}^{d} i x_i^4 + random[0.1] \qquad -1.28 \le x_i \le 1.28 \quad f_{min} = 0 \quad d = dimension$$

B. Multimodal benchmark functions

F8:

$$F_8(x) = \sum_{i=1}^{d} -x_i \sin\left(\sqrt{|x_i|}\right) \qquad -500 \le x_i \le 500 \quad f_{min} = -12\,569.5 \quad d = dimension$$

F9:

$$F_9(x) = \sum_{i=1}^{d} [x_i^2 - 10\cos(2\pi x_i) + 10] \qquad -5.12 \le x_i \le 5.12 \quad f_{min} = 0 \quad d = dimension$$

F10:

$$F_{10}(x) = -20exp\left(-02.\sqrt{\frac{1}{d}\sum_{i=1}^{d} x_i^2}\right) - exp\left(\frac{1}{d}\sum_{i=1}^{d} \cos(22\pi x_i)\right) + 20 + \varepsilon \qquad -32 \le x_i \le 32 \quad f_{min} = 0 \quad d = dimension$$

F11:

$$F_{11}(x) = \frac{1}{400}\sum_{i=1}^{d} x_i^2 - \prod_{i=1}^{d} cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \qquad -600 \le x_i \le 600 \quad f_{min} = 0 \quad d = dimension$$

F12:

$$F_{12}(x) = \frac{\pi}{30}\left\{10sin(\pi x_i) + \sum_{i=1}^{d-1} (x_i - 1)^2\,[1 + 10sin^2(\pi x_{i+1})] + (x_d - 1)^2\right\} + \sum_{i=1}^{d} u(x_i.10.100.4) \qquad -50 \le x_i \le 50 \quad f_{min} = 0 \quad d = dimension$$

F13:

$$F_{13}(x) = 0.1\left\{sin^2(3\pi x_i) + \sum_{i=1}^{d-1} (x_i - 1)^2\,[1 + sin^2(3\pi x_i + 1)] + (x_d - 1)^2 \times [1 + sin^2(2\pi x_d)]\right\} + \sum_{i=1}^{N} u(x_i.5.100.4) \qquad -50 \le x_i \le 50. \; f_{min} = 0 \quad d = dimension \quad where.\, x_i = 1 + \frac{x_i + 1}{4} \quad u(x_i.a.k.m) = \begin{cases} k(x_i - a)^m & x_i > a \\ 0 & -a < x_i < a \\ k(-x_i - a)^m & x_i < -a \end{cases}$$

C. Fixed-dimension multimodal test functions

F14:

$$F_{14}(x) = \left(\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^{2}\left(x_i - a_{i,j}\right)^6}\right)^{-1} \qquad -65.536 \le x_i \le 65.536. \quad f_{min} \approx 1 \quad d = 2$$

F15:

$$F_{15}(x) = \sum_{i=1}^{11} \left[a_i - \frac{x_i(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4}\right]^2 \qquad -5 \le x_i \le 5. \quad f_{min} \approx 0.0003075. \quad d = 4$$

F16:

$$F_{16}(x) = 4x_1^2 - 2.1\,x_1^4 + \frac{1}{3}x_1^6 + x_1 x_2 - 4\,x_2^2 + 4x_2^4 \qquad -5 \le x_i \le 5. \quad f_{min} = -1.0316285. \quad d = 2$$

F17:

$$F_{17}(x) = \left(x_2 - \frac{5.1}{4\pi^2}\,x_1^2 + \frac{5}{\pi}x_1 - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)cos x_1 + 10 \qquad -5 \le x_1 \le 10.\; 0 \le x_2 \le 15. \quad f_{min} = 0.398. \quad d = 2$$

F18:

$$F_{18}(x) = [1 + (x_1 + x_2 + 1)^2\,(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)] \times [30 + (2x_1 - 3x_2)^2 \times (18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)] \qquad -2 \le x_1 \le 2. \quad f_{min} = 3. \quad d = 2$$

F19:

$$F_{19}(x) = -\sum_{i=1}^{4} c_i \exp\left(-\sum_{j=1}^{3} a_{i.j}\left(x_j - p_{ij}\right)^2\right) \qquad 0 \le x_j \le 1. \quad f_{min} = -3.86. \quad d = 3$$

F20:

$$F_{20}(x) = -\sum_{i=1}^{4} c_i \exp\left(-\sum_{j=1}^{6} a_{i.j}\left(x_j - p_{ij}\right)^2\right) \qquad 0 \le x_j \le 1. \quad f_{min} = -3.32. \quad d = 3$$

F21:

$$F_{21}(x) = -\sum_{i=1}^{5}[(X - a_i)(X - a_i)^T + c_i]^{-1} \qquad 0 \leq x_j \leq 10. \qquad f_{min} = -10.1532. \qquad d = 4$$

F22:

$$F_{22}(x) = -\sum_{i=1}^{7}[(X - a_i)(X - a_i)^T + c_i]^{-1} \qquad 0 \leq x_j \leq 10. \qquad f_{min} = -10.4028. \qquad d = 4$$

F23:

$$F_{23}(x) = -\sum_{i=1}^{10}[(X - a_i)(X - a_i)^T + c_i]^{-1} \qquad 0 \leq x_j \leq 10. \qquad f_{min} = -10.536. \qquad d = 4$$