Received: 17 January 2020 | Revised: 1 July 2021 | Accepted: 3 July 2021
DOI: 10.1002/cpe.6514

RESEARCH ARTICLE

WILEY

Chaos teaching learning based algorithm for large-scale global optimization problem and its application

Alok Kumar Shukla

Department of Computer Science and Engineering, Vellore Institute of Technology (VIT)-AP University, Amaravati, India

Correspondence
Alok Kumar Shukla, Department of Computer Science and Engineering, Vellore Institute of Technology (VIT)-AP University, Amaravati, Andhra Pradesh, India.
Email: alokjestshukla@gmail.com

Abstract

Teaching learning-based optimization (TLBO) is a popular stochastic algorithm that has recently been widely applied in a variety of optimization problems since its start. In TLBO algorithm, the concept of chaos not only shows a vital effect in its convergence but also plays a substantial role to balance of exploration and exploitation through evolution. However, TLBO is quickly trapped in local optima and premature convergence seems when applied to sophisticated complex functions. To handle these problems, we introduced an improved TLBO algorithm using chaotic concept. To achieve ability to search for exploration and exploitation, new phase called chaotic phase is added in original TLBO algorithm. The proposed method is thoroughly evaluated on benchmark test suites. The numerical result show that proposed method is relatively effective in adapting the chaotic value regarding original TLBO in terms of solution quality and convergence rate. In addition, performance of proposed method is evaluated on benchmark KDD Cup 99 intrusion dataset. The experimental results demonstrate that proposed method achieves higher predictive accuracy, detection rate, false alarm rate, and provided more significant features as compared with other wrapper techniques.

KEYWORDS
accuracy, chaotic, KDD cup 99, optimization, teaching learning-based optimization

1 | INTRODUCTION

In general, the process of selecting the best available of several possible decisions (options) is called optimization. The best possible decision (or alternative), possible circumstances (i.e., either minimum or maximum) is called optimal solution.[1] In broader scene, researcher coarsely define combinatorial optimization problems as problems that cannot be resolved with optimality or with any limit given by standard techniques within a reasonable period.[2] These problems can be separated into different groups depending on whether they are constant or discrete, restricted or not, and single/multi-objectives.[3] Finding the optimal global value of a function has become more difficult or practically intolerable for several problems. Therefore, more resourceful optimization algorithms are generally required to solve the complex type of problems such as numerical and intrusion detection problems.

In the nature of optimization, several solution techniques have been obtainable for the different types of optimization problems. These techniques can be broadly classified into two main categories: exact and stochastic methods.[4] In general, an exact methods are usually numerical and deterministic (search or optimization algorithm is said to be deterministic if it consistently follows the same execution path [and returns the same solution] when it is repeatedly run from the same starting solution) whereas stochastic methods are generally population-based stochastic (it is the opposite of deterministic methods and are therefore probabilistic (random). They may not follow the same execution path (or return the same solution) when repeatedly run from the same starting solution) strategies.[5] In addition, stochastic methods are further classified in two ways such as heuristic and metaheuristic methods.[6] The term heuristic is the practice of finding the near-optimal solution by trial-and-error method whereas metaheuristic is independent from problem specification.[7]

From continuous domain, global optimization area has remarkably robust, making different types of deterministic and stochastic algorithms for optimization in recent decades.[8] In context of stochastic methods, evolutionary techniques deal with large number of exclusive advantages such as toughness, reliability, global research capability, and minimum or no information requirements.[9] These features of evolutionary algorithms (EAs) have additional benefits such as effortless implementation, and parallelism; make it an interesting alternative. As a result, there have been enormous amount of literature concerning the optimization of practical problems can be solved by EAs.[10]

Due to the shortcomings of traditional algorithms such as local solution and stagnation, research on swarm intelligence-based algorithms have become hypothetical hotspot in current decades. It is noted that EAs have generally been mentioned as effective methods for complex optimization problems, primarily due to their implicit parallelism which enables concurrent convergence towards the entire set of space. Despite their gaining popularity, a common challenge arising in the use of EAs, and other population-based metaheuristic methods, is the need to conduct a large number of function evaluations before a set of near-optimal solutions can be achieved thus need to develop the powerful population-based metaheuristic algorithms in recent decades.[11] One of the most important benefits of EAs is leveraged the balance between exploration and exploitation and overcome the premature problem. In addition, metaheuristic algorithm finds the optimal solution from local search capabilities with global searching capabilities for standard test suites and intrusion detection problems, for example, genetic algorithm (GA), particle swarm optimization (PSO), taboo search, and teaching learning-based optimization (TLBO).[12,13]

TLBO method addresses difficult tasks equivalent to other stochastic algorithms. Like difficult multimodal tasks happen to be relieved, standard TLBO method can always be cornered in the local optimal, and convergence speed definitely decreases noticeably in any last cycle of evolution. However, it is not always possible to correctly implement a global search. Therefore, in some effects, TLBO cannot find the optimal global solution. The search method used in the basic TLBO is generally based on learning capability. Therefore, it is not always possible to treat the problem correctly. So, there is great amount of literature that attention on refining the performance of TLBO algorithm by increasing exploration and exploitation search ability. Through TLBO, exploration and exploitation are certain by way of the adaptive values of the chaotic which can increase the learner's diversity in the algorithm.

With the growth of nonlinear dynamics, chaos theory has been extensively used to increase both convergence rate and solution quality.[14] So far, chaos theory has been successfully shared with different metaheuristic algorithms. Some primary efforts in this area includes PSO, GA, GOA, hybrid of chaotic sequences using differential evolution, cuckoo search, artificial immune algorithm, gravitational search algorithm, and biogeography-based optimization.[15] Influenced by chaotic map theory, this article proposed an improved variant of TLBO using 10 chaotic maps. To obtain a considerable better balance involving the exploratory and exploitative tendencies, we have included chaotic map in TLBO. Main goal is to update the solution of quality by using various chaotic maps. In order to reach this task, various TLBO algorithms which work chaotic maps as a proficient substitute for linear decrement series have been proposed.

To overcome the limitation of traditional algorithms, in this study we uses chaotic functions in TLBO to rise the perturbation power which can help the learner jump of the local optimal solution but convergence of the algorithm is significantly reduced. Therefore, how to modify the random approach to find the best balance of disturbance intensity is a demand that is value considering. By this way, it is intended to accelerate the global convergence rate and to prevent the TLBO from getting local optima. The key input of this article are:

- The chaotic-map is employed to produce uniformly distributed learners to increase the population diversity.
- Random topological direction is incorporated by chaotic-map, therefore neighborhood learner improves their search capability.
- Obtains the more efficient results for optimization problems and compared with 10 variants of chaotic maps.
- The performance of proposed method is tested on benchmark test functions.
- Chaotic TLBO is also active on anomaly detection using KDD Cup 99 dataset. In addition, proposed method selects small number of features and is less prone to overfitting than using whole features.

This study is the first of its kind that uses the different chaotic map variants in the basic TLBO for solving global optimization and intrusion detection system (IDS) problems. The experimental result validates that chaotic map is used for global capabilities while existing insignificant tuning parameter performs local ability. When we change the value of the chaotic value, the search space may change dynamically. The experiments shown remarkable results in both the speed of convergence and the ability of finding global optima in used benchmark functions and also on KDD Cup 99 dataset thus proving a better balance between exploration and exploitation in proposed compared with TLBO.

The remainder of this article is organized as follows. Section 2 provides literature review of metaheuristic based algorithms. Section 3 discussed the categorization of the different type of chaotic map and its parameters are also discussed. In Section 4, we presented the chaotic stage in basic TLBO. In Section 5, numerical experiments are performed to test the proposed algorithm with the other kinds of the chaotic map as well as other EAs on well-known test suites for the global optimization. Section 6 shows the behavior of the proposed method in IDS. Finally, conclusions are drawn in Section 7.

2 | RELATED WORK

A comprehensive study with ancient details of global optimization problems can found in.[16] Since this article aims to investigate global optimization problems and propose solutions from the global perspective. Basically, this article focuses on the population-based stochastic optimization frameworks under the paradigm of metaheuristic computation. The main benefits of metaheuristic algorithms are; simple and easy to use and robust on the situation where the cost accomplishment is dynamic or noisy.[17] It is the process of acquiring the current information through some criteria to help significantly it to determine how accurately to generate the next candidate solution or which solutions should be processed next. The normally uses statistical details achieved from samples from the search space or some hypothetical versions from natural phenomena or physical procedures. Nowadays, several scholars have applied TLBO in solving optimization problems.[18] As a result, some TLBO amendments were also investigated to enhance its performance regarding the convergence rate. Thus, innovative methods that can help to give better balance among exploration and exploitation in TLBO.[19]

Teaching–learning-based optimization is an algorithm without any effort for fine tuning initial parameters which has fast convergence speed while it is easy to fall into local optimum in solving complex global optimization problems, originally which has no value of chaos.[20] For solving complex engineering optimization problems, seeing the features of neural network algorithm and TLBO, an effective hybrid method, called TLNNA was proposed.[21] The performance of TLNNA for 30 well-known unconstrained benchmark functions and four engineering optimization problems was examined and the optimization results were compared with other competitive EAs.

Similar to other typical EA methods, TLBO is also trapped in local optimum when solving complex problems with multiple local optimal solutions. The referred literature aim is to achieve the balance between local and global search which have not been contract an optimal solution. To solve global optimization problems, in Reference [22] introduced a modified TLBO (mTLBO) and performance of its compared with state-of-the art forms of PSO, DE, and ABC algorithm. In addition, proposed method investigated the data clustering performance of mTLBO over other EAs on few standard synthetic and artificial datasets.

In Reference [23], author introduced new TLBO variant based on neighbor learning and differential mutation. The concept of neighbor learning and differential mutation was announced to improve the convergence solution after each run of experiment. This method maintained the explorative and exploitation search of population and disappoints the premature convergence. The efficiency of the proposed algorithm was evaluated on eight benchmark functions of Congress on Evolutionary Computation 2006. Furthermore, Birashk et al.[24] investigated a dynamic multiobjective optimization method that tried to detect changes and track them, using the knowledge of prior environments to converge to the new Pareto-optimal front more quickly. In this study, a cellular automata-based approach was first proposed for managing and evaluating solutions during the optimization process.

According to Taher Niknam and Golestaneh,[25] θ-multiobjective TLBO algorithm was useful to resolve the complications of dynamic communications. It is applied for choosing the best learner, and an optimal Pareto-frontal face distributed evenly. On the other hand, by Murty et al.[26] has introduced the automatic concept clustering on substantial unlabeled datasets. Most of the topical clustering methods, the proposed algorithm were not require preliminary information about the dataset for correctly classified relatively measures the number of partitions. In Reference [27] author introduced a novel bio-inspired optimization algorithm, namely, barnacles mating optimizer (BMO) to solve optimization problems. The proposed algorithm mimics mating behavior of barnacles in nature for solving optimization problems. First, BMO was benchmarked on a set of 23 mathematical functions to test characteristics of BMO for finding the optimal solutions. Then, it was applied to optimal reactive power dispatch problem to verify the reliability and efficiency of BMO.

A novel hybrid method based on adaptive neuro-fuzzy inference system optimized with a combination of GA and teaching-learning-based optimization algorithm is employed.[28] The input parameters were latitude, longitude, design point DNI and SM, while the annual energy produced, livelier cost of energy and capacity factor are the target variables. The results of the study show that although the annual energy produced by SPTS rises by increasing the SM and decreasing design point DNI, optimum design parameters should be determined by the economic factors. Similarly, in Reference [29] used the support vector machine for an advanced structural risk minimization based learning system. This work is extended to feature selection problems and also used to solve the multiobjective gene selection problems.

However, in Reference [30] author have presented improved TLBO and mutated fuzzy adaptive PSO method to find the tiny subset of genes in breast cancer regarding classification accuracy. The teaching learning based optimization algorithm suffered with premature convergence and lack of trade-off between local search and global search. Hence, to addressed still shortcomings of TLBO, new chaotic version of TLBO algorithm was proposed with different chaotic mechanisms.[31] Further local search method was also incorporated for effective trade-off between local and global search and to improve the quality of solution. The performance of proposed algorithm was evaluated on some benchmark test functions taken from Congress on Evolutionary Computation 2014. Consequently, innovative ideas and methods that might help give better balance among pursuit and exploitation in TLBO are necessary.[19] This necessity determined you to develop a tremendous TLBO algorithm structured about new advanced concepts/strategies. In Reference [32] presented a feature reduction method using merging ranks that obtained from information gain and correlation-based methods to classify normal and abnormal attack.

A new SSA-based method (MCSSA) is proposed that performed chaotic exploitative trends and has a multipopulation structure.[33] The new structure can assist salp swarm algorithm (SSA) for making a more stable trade-off between global exploration and local exploitation capabilities.

The exploitation trends and neighborhood searching commands of SSA were enriched using chaos-assisted exploitation strategy. Then arranged multipopulation structure with three substrategies to enlarge the global exploration capabilities of the algorithm. To test the performance of proposed MCSSA, a set of comprehensive algorithms was used, including other original methods, conventional SSA, and 13 advanced techniques based on 30 IEEE CEC 2017 benchmark functions and five IEEE CEC 2011 practical test problems.

The chaotic gravitational search algorithm used chaotic maps for improving diversity to solve global optimization problem, it still has problems with the balance of exploration and exploitation. In Reference [34] author proposed balance adjustment based chaotic gravitational search algorithm introduced the sine randomness function and balance mechanism to solve above-mentioned problem. Similarly, chaotic maps for parameter selection concept have been applied and design a novel feature selection approach through.[35] Each time a random number is required to the original negative selection for mutation and generation of the initial population over and done with chaotic number generators in this study. In the reported literature,[36] several anomaly or fault-detection approaches shown the training data have not represented all normal data and self, and nonself-space often vary over time. At that time, they have introduced the unsupervised clustering process based on rising hierarchical self-organizing maps including unit labeling and efficient determination of the winning unit. In Reference [37], a novel dimensionality reduction technique was introduced to find the optimal numbers of features using NB classifier for IDS dataset which generated striking feature subsets.

3 | EXISTING CHAOTIC MAP STRATEGIES

In general terms, chaos is an exact in nature, randomness in nonlinear, and bounded to solve the complex structure[8] can see in Figure 1. Generally, chaos enhanced the optimization quality when it have qualities such as stochasticity, understanding to the initial conditions, and periodicity, in addition, it transformed into chaotic maps. The landscape of chaos is definitely random and irregular, and also it has a component of regularity can be seen in Figure 2. Chaos uses chaotic variables instead of random variables. Consequently, it can perform absolute searches at sophisticated speeds compared with the inexact search that primarily relies on probabilities.[39] Moreover, it plays a significant role in improving EAs to avoid the local optima and speed-up the convergence rate. Simple scarce functions and some restrictions are needed actually for long sequences.[40] In addition, a massive quantity of diverse sequences can be produced by merely changing its initial condition.

As we know that chaos value is an essential part of the balance between global and local search. From the position of mathematical analysis, we understood that chaos maps strongly influence the TLBO performance. The fundamental objective is to realize the merits and demerits of each chaos to formulate more active strategies for TLBO algorithm. A wide variety of chaotic maps is available in optimization fields. In this section, the 10 most widely used chaotic maps have been used.[41] The mathematical modulations of these chaotic maps used are described as follows:

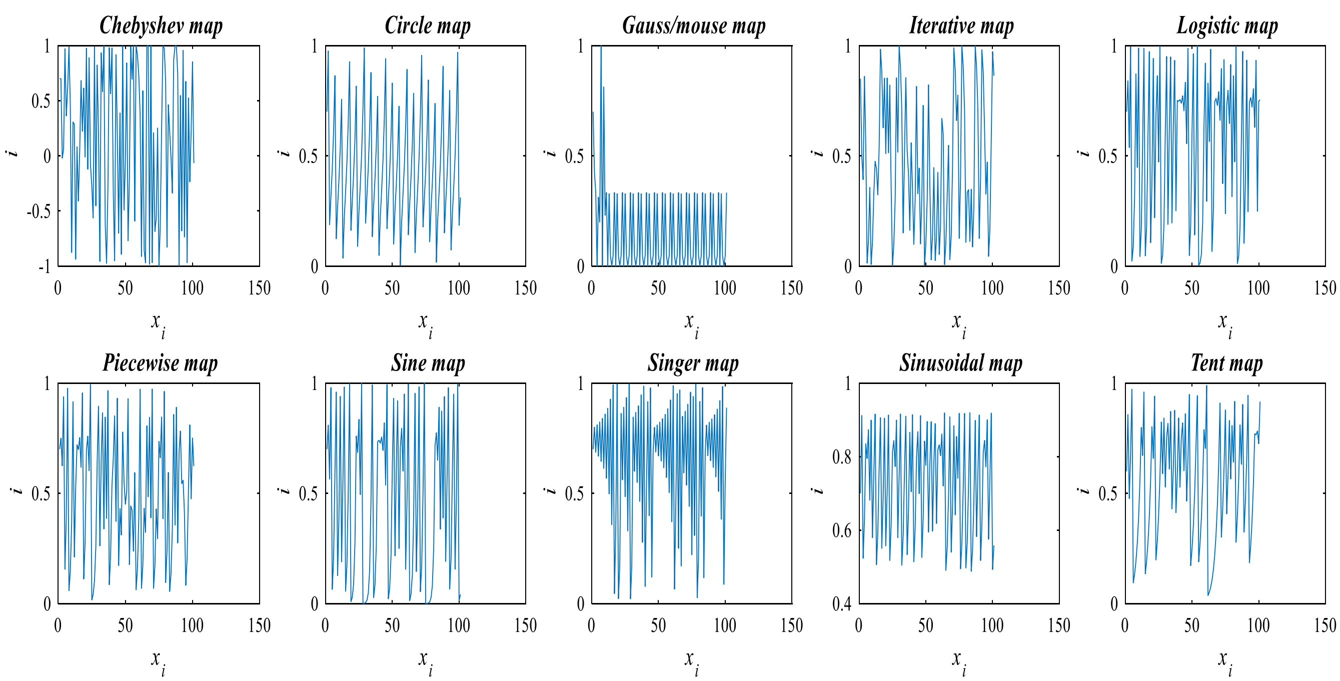

FIGURE 1 Visualization of chaotic maps

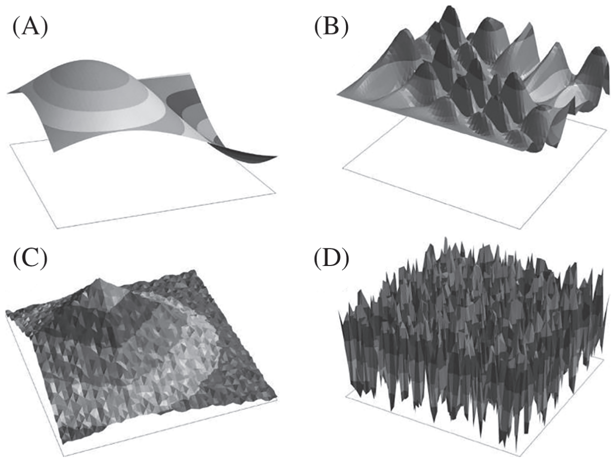

FIGURE 2 Fitness landscapes with four different ruggedness properties: (A) smooth landscape, (B) locally smooth/globally rugged landscape, (C) locally rugged/globally smooth landscape, and (D) highly rugged landscape (with random heights)[38]

1. Logistic map

The equation of the logistic map is expressed according to Equation (1).

$$x_{i+1} = l.x_i(1 - x_i), \tag{1}$$

where, l represent as control parameter set as 3.4 to generate numbers between (0, 1).

2. Circle map

It is a one-dimensional map which is a member of dynamical systems on circle originally Andrey Colmogorov have presented can be defined as Equation (2).

$$x_{i+1} = \left[x_i + b - \left(\frac{l}{2\pi}\right)\sin(2\pi x_i), 1\right] \mathrm{mod}(l), \tag{2}$$

where l represent as control parameter, generates chaotic numbers lies between 0 and 1 by using $l = 0.5$ and $\mathrm{b} = 0.2$.

3. Gauss map

Gauss map is expressed as following Equation (3).

$$x_{i+1} = \begin{cases} 1 & x_i = 0 \\ \frac{1}{\mathrm{mod}(x_i,1)} & \text{otherwise} \end{cases} . \tag{3}$$

This map also generates chaotic sequences in (0, 1).

4. Iterative map

The iterative chaotic map is shown in Equation (4).

$$x_{i+1} = abs\left(\sin\left(\frac{l\pi}{x_i}\right)\right), \tag{4}$$

where l is an adjustable parameter.

5. Chebyshev map

Chebyshev map is expressed as Equation (5).

$$x_{i+1} = \cos(i.\cos^{-1} x_i). \tag{5}$$

6. Piecewise map

The formula of piecewise map is expressed in order to Equation (6).

$$x_{i+1} = \begin{cases} \frac{x_i}{l}, & 0 \le x_i < l \\ \frac{x_i - l}{0.5 - l}, & l \le x_i < 0.5 \\ \frac{1 - l - x_i}{0.5 - l} & 0.5 \le x_i < 1 - l \\ \frac{1 - x_i}{l} & 1 - l \le x_i < 1 \end{cases},$$

where, l represent the control parameter which lies between 0 to 0.5.

7. Sine map

It is defined as following Equation (7).

$$x_{i+1} = \frac{p}{4}\sin(\pi x_i), \tag{7}$$

p represents as control parameter having values in the range (0, 4).

8. Singer map

The singer map is characterized as Equation (8).

$$x_{i+1} = l(7.86x_i - 23.31x_i^2 + 28.75x_i^3 - 13.30x_i^4), \tag{8}$$

where, $l = 1.07$ and p represents as control parameter lie between (0.9, 1.08).

9. Sinusoidal map

This map is formulated as follows Equation (9).

$$x_{i+1} = l.x_i^2 \sin(\pi x_i), \tag{9}$$

where l is the control parameter set as 2.4.

10. Tent map

The equation of the tent map can be represented as Equation (10).

$$x_{i+1} = \begin{cases} \frac{x_i}{0.7}, & x_i < 0.7 \\ \frac{10}{3}(1 - x_i), & x_i \ge 0.7 \end{cases}. \tag{10}$$

4 | CHAOTIC TEACHING LEARNING BASED OPTIMIZATION

TLBO is a recently powerful population-based algorithm investigated by Rao et al. based on teaching-learning phenomena. It works on the concept of classroom learning paradigm. In this learning, teachers are available to teach the learners and their aim is to improve the learning capability of the learners. But, in classroom learning paradigm, a learner can also enhance its skill by acquiring the knowledge from other learners.[22] It depends on the acquired knowledge through the teacher and learns from the classmates.[42] The group of learners also called class.

The working structure of TLBO depends on the enormously sophisticated learners so that it produces better grades. They learn through neighbor learning phenomena. Similar to other EAs, TLBO can improve the solution quality of learners in order to get the optimal global solution. Characteristically, individuals show a collection of learners where each consists of the variables to be optimized. TLBO works can be elaborated in two stages: teacher phase and learner phase. In teacher phase, gets with best learner in the form of teacher, and learner phase concerns how learning is carried out through the interaction between learners. The position of the *i*th learner is presented as:

$$X_{i,k} = \{X_{i,1}, X_{i,2}, \dots, X_{i,D}\}$$

where, L_b is lower bound and U_b is upper bound of *D* dimension in the search space $X_{i,D} \in [L_b, U_b]$. The learner *X* is randomly adjusted within search space capabilities. The evolution of $X_{i,k}$ is randomly generated by the following Equation (11).

$$X_{i,k} = L_b + r_1 * (U_b - L_b), \tag{11}$$

where, $i = 1, 2, 3 \dots \text{nPop}$, $k = 1, 2, 3, \dots, D$, r_1 means random variable, L_b shows the lower bound, and U_b signifies the upper bound value. The simulation of classical learning process to all learners are categories into teacher phase and learner phase. The best learner is obtained from the teacher phase through knowledge learning from the neighbors, on the other hand, learner phase finds the best learner through interaction between different learner groups. The pseudocode of the chTLBO algorithm is presented in Algorithm 1. The teacher phase and learner phase are presented as follows:

4.1 | Teacher phase

The main aim of this phase is to conveying the student knowledge such that results of class improve significantly and can lead the mean result of class. In general, the teacher can improve the result up to certain level. In practice, several constraints are responsible for results, such as teaching method, professors capability, learners selfish ability, interaction of learners to others and knowledge of learners. Moreover, teacher can increase the mean result of class to certain value which depends on the capability of the whole class.

Let $M_{i,k} = (1/\text{nPop})\,(\Sigma X_{i,k})$ be mean value of particular topic where $k = 1, 2, \dots, D$. The updating equation of process as described in Equation (12).

$$X_{i,k}^{\text{new}} = X_{i,k}^{\text{old}} + r_2 * (X_{\text{teacher},k} - T_f * M_{i,k}) \text{ and } T_f = \text{round}[1 + \text{rand}(0, 1)]. \tag{12}$$

Here, $X_{\text{teacher},k}$ is the best learner at current iteration of algorithm, r_2 is the random numbers lies between 0 and 1; T_f represents as teaching factor. In each iteration, $X_{i,k}^{\text{new}}$ is the updated from the value of $X_{i,k}^{\text{old}}$. $X_{i,k}^{\text{new}}$ and $X_{i,k}^{\text{old}}$ denotes the *k*th learners select after or before learning from the good learner (as teacher).

4.2 | Learner phase

The second part of this process is learner phase, increases the knowledge of learners by the two dissimilar ways: In first approach, input from the best learner (teacher), and next way as through mutual interaction between them. The goal of each learner is randomly interacted with peer learners and enhances communication grade. To opt *i*th learner as X_p and another random learner is X_q(where p≠q) by mutual interaction with learners. The new solution of the ith learner X_p in the learning phase is described as Equations (13) and (14).

For i = 1: nPop.
Randomly select two learners X_p and X_q
if $(f(X_p) < f(X_q))$

$$\text{new}X_i = \text{old}X_i + r * (X_p - X_q) \tag{13}$$

else

$$\text{new}X_i = \text{old}X_i + r * (X_q - X_p) \tag{14}$$

end if
end For

where *r* represents the uniformly random number lies between 0 and 1, $f(X_p)$ and $f(X_q)$ are the Best solution of the learners X_p and X_q. In view of size of learner group is *nPop*, where learner communicates from a good learner to acquire knowledge. Further details will be processed sequentially in the following subsections.

4.3 Chaos-stage

To create an effective trade-off between exploration and exploitation capabilities, in the available literature, enormous number of chaos-based algorithms have been introduced which has satisfied attention to solve real-world optimization problems.[43] In general, the generation of populations in the search space acquires an important role in TLBO. Based on this response, we effort to solve the composite issue by using the initialization of the chaotic-map into standard TLBO.

Chaos is a deterministic, random-like mathematical phenomena which takes place in nonlinear system and it is toughly affected by the initial conditions. By changing initial values, very long and different chaotic sequences can be generated to diversify students and improve the performance of TLBO to avoid premature convergence. Chaotic sequence is produced by a chaotic map which is generally formulated on the concept of the following steps:

Algorithm 1. Chaotic teaching learning-based optimization (chTLBO)

Begin
  Select nPop (population), D (dimension), and $t_{\max}$ (maximum iterations)
  Randomly generate the value of learners and
  Evaluate the cost value for each learner
  Calculate $X_{\text{mean}} = \sum_{i=1}^{\text{nPop}} \frac{x_t}{\text{nPop}}$
  Determine best learner (X_{teacher})
  While ($t < t_{\max}$)
      $t = t + 1;$
      Diff_Vector = ($X_{\text{teacher}} - T_f{}^*X_{\text{mean}}$)
    For $i = 1$ to nPop
      For $k = 1$: D
      $X_{i,k}^{\text{new}} = X_{i,k}^{\text{old}} + \beta(t) * \text{Diff\_Vector}(k)$
      Calculate cost value of each learner ($X_{i,k}^{\text{new}}$)
    End For
      if $f(X_{i,k}^{\text{new}}) < f(X_i)$
      $X_i = X_{i,k}^{\text{new}}$
      End if
  End For
  For $i = 1$ to nPop
    Randomly select the other two learners as X_p and X_q
    For $k = 1$: D
      if $f(X_p) > f(X_q)$
        $X_{i,k}^{\text{new}} = X_i + r * (X_p - X_q)$
      else
        $X_{i,k}^{\text{new}} = X_i + r * (X_q - X_p)$
      End if
  End For
  Estimate cost value of each learner as $X_{i,k}^{\text{new}}$
      if $(f(X_{i,k}^{\text{new}}) < f(X_i))$
      $X_i = X_{i,k}^{\text{new}}$
      End if
  End For
      $X_{i,k}^{\text{new}} = X_{i,k} + \beta(2X_i - 1)$
      if $(f(X_{i,k}^{\text{new}}) < f(X_i))$
      $X_i = X_{i,k}^{\text{new}}$
      End if
    Determine Best solution in entire population
  End while
End

Stage 1: Outlining chaotic variable for next generation using chaotic methods (see in Section 3).

Stage 2: Defining upper and lower boundaries of η(range factor) by Equations 15 and 16, to preserve the new decision variable within their boundaries.

$$\eta_{\max} = \frac{(X_{\max} - X_{\text{new}})}{(2x_i - 1)}, \tag{15}$$

$$\eta_{\min} = \frac{(X_{\min} - X_{\text{new}})}{(2x_i - 1)}, \tag{16}$$

where, $X_{\max}$ and $X_{\min}$ are upper and lower boundaries of the decision vector and $\eta_{\max}$ and $\eta_{\min}$ are maximum and minimum of range factor η.

Stage 3: Select η according to following Equation (17).

$$\eta = \frac{\eta_{\max} + \eta_{\min}}{2} \tag{17}$$

Stage 4: Assess new decision values $X_{i,k}^{\text{new}}$ by converting chaotic values x_i by Equation (18).

$$X_{i,k}^{\text{new}} = X_{i,k}^{\text{old}} + \beta(2x_i - 1) \tag{18}$$

Stage 5: Evaluate the new correctness for original decision values $X_{i,k}^{\text{new}}$

Stage 6: If the new solution $X_{i,k}^{\text{new}}$ is better than previous solution, in terms of fitness value, output the new solution as result of proposed, otherwise, let $k = k + 1$ and go to the Teacher phase.

For example, in the logistic map, chaotic variable x_i ε (0, 1) conditions that the original leaves aside x_0 ε (0, 1) from detailed periodic static points (0, 0. 25, 0. 5, 0. 75, 1). μ is certainly a fixed continuous set of 4, also known as bifurcation coefficient. When the value of μ rises from zero, the powerful system produced by equations to changes of a fixed one indicated as two and up to 2^n. In this procedure, large number of manifold periodic elements will be located in the thinner and thinner μ intervals as it increases. This phenomenon is really without restrictions. But it includes limit value at $\mu_t = 3.60$. Remember that when the techniques are, the period μ can be infinite or even nonperiodic μ_t. For the moment, the whole structure evolves into chaotic state. However, when μ is larger than 4, whole system becomes unstable. Therefore, interval [μ; 4] is commonly measured by the chaotic area of the whole system. It is well-defined as follows Equation (19).

$$x_{i+1} = \beta(x_i, \mu) = \mu x_i(1 - x_i). \tag{19}$$

The bifurcation diagram is certainly note that the central element of logistic-map initialization is to create the identical chaotic variables that correspond to the optimization problem. More obviously, whenever a current quantity of chaotic generations is executed, the chaotic variables will be produced accordingly. Subsequently, by remapping these variables in the optimization space, the preliminary variables will be generated for the initial optimization problem. The logistic-map pseudocode works as following in Algorithm 2, which produces a uniformly distributed sequence and prevents it from being immersed in small periodic cycles effectively.

Algorithm 2. Logistic map for initialization

```
Begin
  Randomly initialize chaotic vector;
  While(condition not meet)
    If(chaotic vector plunges into fixed points and small periodic cycle)
      Implement the small positive random perturbation;
      Map them according to Equation (19);
    else
      Update vector according to Equation (19) directly;
    End if
      Next iteration until stopping;
      Remap the chaotic vector into search space;
  End while
End
```

5 | EXPERIMENTAL RESULT AND ANALYSIS

Entirely experimental data are coded in the MATLAB R2016a environment under Windows 8 operating system. All simulations are running on a computer with Intel Core (TM) i5 CPU @ 2.60 GHz with 8 GB of memory. In order to test the performance and efficiency of the proposed method, a variety of experiments are conducted. Due to the stochastic nature of the optimization algorithm, we employ appropriate and sufficient sets of test functions and case studies to ensure that the superior results of the algorithm do not happen by chance. The stopping condition of algorithms is maximum number of iterations and maximum independent runs for better solution. The EAs run on benchmark problem until corresponding stopping criteria meet.

5.1 | Benchmark test functions

In order to test the performance of proposed algorithm, variety of experiments have conducted on popular test suites carried out from References [44,45] applied to assess the performance of methods. The behavior of all benchmark problems is minimization of corresponding problem and find out the optimal global solution. In first experiment, we used seven optimization problems as unimodal notation (f_1–f_7) and while remaining are multimodal optimization problems. Tables 1–3 represents the unimodal and multimodal test functions. Our objective is moving towards symmetric search space by chaotic approach on the benchmark problems.

TABLE 1 Standard seven unimodal benchmark problems

F. No.	Function	Equation	Range
F1	Sphere	$F\ \min = \sum_{i=1}^{D} x_i^2$	[−100, 100]
F2	Schwefel 2.22	$F\ \min = \sum_{i=1}^{D} \mid x_i \mid + \prod_{i=1}^{D} \mid x_i \mid$	[−10, 10]
F3	Schwefel 1.2	$F\ \min = \sum_{i=1}^{D} \mid x_i \mid \left(\sum_{j=1}^{D} \mid x_j^2 \right)^2$	[−100, 100]
F4	Schwefel 2.21	$F\ \min = \max_i\{\lvert x_i \rvert, 1 \le i \le n\}$	[−100,100]
F5	Rosenbrock	$F\ \min = \sum_{i=1}^{D} [100(x_i^2 - x_i + 1)2 + (1 - x_i)2]$	[−30, 30]
F6	Step	$F\ \min = \sum_{i=1}^{D} [x_i + 0.5]^2$	[−100, 100]
F7	Quartic	$F\ \min = \sum_{i=1}^{D} i x_i^4 + \text{rand}\ (0, 1)$	[−1.28, 1.28]

TABLE 2 Standard six multimodal benchmark problems

F. No.	Function	Equation	Range
F8	Schwefel	$F\ \min = -\sum_{i=1}^{D} (x_i \sin(\sqrt{\mid x_i}))$	[−500, 500]
F9	Rastrigin	$F\ \min = \sum_{i=1}^{D} [x_i^2 - 10\cos(2\pi x_i) + 10]$	[−5.12, 5.12]
F10	Ackley	$F\ \min = -20\ \exp\left(-0.2\sqrt{\frac{1}{D}\sum_{i}^{D} x_i^2}\right) - \exp\left(\frac{1}{D}\sum_{i=1}^{D} \cos\ 2\pi x_i\right) + 20 + e$	[−32, 32]
F11	Griewank	$F\ \min = \frac{1}{4000}\sum_{i=1}^{D} x_i^2 - \prod_{i=1}^{D} \lVert \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	[−600, 600]
F12	Penalized 1	$F\ \min = \pi/n\{10\sin^2(\pi y_i) + \sum_{i=1}^{n-1}(y_i - 1)^2 [1 + 10\sin^2(\pi y_{i+1})] + (y_n - 1)^2\} + \sum_{i=1}^{n} u(x_i, 10, 100, 4), y_i = 1 + \frac{1}{4}(x_i + 1)$ $u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m, & x_i > a, \\ 0, -a \le x_i \le a, & \\ k(-x_i - a)^m, x_i < -a, & \end{cases}$	[−50,50]
F13	Penalized 2	$F\ \min = 0.1\{\sin^2(3\pi x_1) + \sum_{i=1}^{n-1}(x_i - 1)^2[1 + \sin^2(3\pi x_{i+1})] + (x_n - 1)[1 + \sin^2(2\pi x_n)]\} + \sum_{i=1}^{n} u(x_i, 5, 100, 4)$	[−50,50]

TABLE 3 Standard 10 multimodal benchmark problems with low dimension

F. No.	Function	Equation	Range
F14	Foxholes	$F\ \min = \left[\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^{2} (x_i - a_{ij})6}\right]^{-1}$	[−65.536,65.536]
F15	Kowalik	$F\ \min = \sum_{i=1}^{11} \left(a_i - \frac{x_1(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4}\right)^2$	[−5, 5]
F16	6 Hump Camelback	$F\ \min = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - x_2^2 + 4x_2^4$	[−5, 5]
F17	Branin	$F\ \min = \left(x_2 - \frac{5.1}{4\pi}x_1^2 + \frac{5}{\pi}x_1 - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos x_1$	[−5, 10] [0, 15]
F18	Gold Stein-Price	$F\ \min = [1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_{2+}6x_1x_2 + 3x_2^2)]$ $[30 + (2x_1 - 3x_2)^2] * (18 - 32x1 + 12x_1^2 + 48x_2 - 36\ x_1x_2 + 27\ x_2^2)$	[−2, 2]
F19	Hartman 3	$F\ \min = \sum_{i=1}^{4} c_i \exp - \left[\sum_{j=1}^{6} a_{ij}(x_j - p_{ij})^2\right]$	[0, 1]
F20	Hartman 6	$F\ \min = \sum_{i=1}^{4} c_i \exp - \left[\sum_{j=1}^{6} a_{ij}(x_j - p_{ij})^2\right]$	[0, 1]
F21	Shekel 5	$F\ \min = -\sum_{i=1}^{5} [(x - a_i)\ (x - a_i)^T + c_i]^{-1}$	[0, 10]
F22	Shekel 7	$F\ \min = -\sum_{i=1}^{7} [(x - a_i)\ (x - a_i)^T + c_i]^{-1}$	[0, 10]
F23	Shekel 10	$F\ \min = -\sum_{i=1}^{10} [(x - a_i)\ (x - a_i)^T + c_i]^{-1}$	[0, 10]

Due to the simple properties of the global optimum in mentioned Tables 1–3 types of test problems, very few researchers have investigated metaheuristic algorithms that specifically exploit these properties. For example, if the algorithm finds the optimal global values of 1–*D*, the values of all remaining dimensions can be easily obtained by replication the detected value to these dimensions. To avoid the bias property, the shifted functions are extended from existing benchmark functions by moving the global optimum to a random position within the search range.

5.2 | Comparison between TLBO and chaotic TLBO methods

We associate with proposed in terms of convergence speed, optimal solution, and stability. Four criteria are used for these algorithms: Best, mean, average, and SD. These experiments used valuation parameters such as population size (nPop), maximum allowed number iteration (max_iter), and dimension of variables (*D*) to evaluate the performance of methods. The performance evaluation criteria to EAs on benchmark problems are Best solution so far and mean of the Best solution to each independent run. To evaluate the performance, set the maximum number of iterations as 1000, population size as 30, and results of comparative experiments are mean over 30 independent runs to all EAs.

Generally, when we solve the benchmark problems, the fitness evaluation interpretations most time in the TLBO is highly efficient regarding less computational burden. Therefore, in this study, we did not compare the computation times of EAs with proposed method. In addition, main difference between proposed and simple TLBO is the position updating equation using chaotic stage, when we compare the computational complexity of our algorithm then found that it was similar. In the experiment, sequential implementation is used, while it is easy to be modified to a parallel application. With an identical form, the performance is likely to be not affected much while computational efficiency improves. More understandability of algorithms, Tables 4–6 presents the Best solution and average solution in 30 runs of 10 TLBO chaotic variants. Figure 3 shows the convergence curve regarding the best fitness value of the median term of each algorithm for unimodal and multimodal functions.

From the experimental results of proposed method is assessed by 23-real benchmark optimization problems. The detail descriptions of these functions are in Reference [46]. The number of decision variables *D* is set to 30 for all test functions. The performance criteria of proposed method are an average solution and Best solution so far as forsee in Tables 4–6. The experiment is evaluated in standard benchmark test suites. The experimental result demonstrates that the chTLBO1 technique is relatively useful in adapting the value of chaotic in comparison to the other TLBO with regards to the quality of the solution, convergence rate.

TABLE 4 Results of 10 chaotic maps and TLBO on benchmark functions

Methods	Performance	F_1	F_2	F_3	F_4	F_5	F_6	F_7
TLBO	Best solution	8.24E-07	5.61E-05	7.80E-06	3.30E-07	26.43	7.32E-05	0.0163
	Average solution	7.52E-06	5.56E-03	8.65E-04	5.95E-05	22.65	5.66E-04	0.9547
chTLBO1	Best solution	2.89E-08	3.23E-04	2.49E-07	2.48E-05	15.63	2.98E-07	0.0031
	Average solution	1.21E-07	1.22E-04	1.61E-05	1.51E-04	19.31	1.62E-06	0.4861
chTLBO2	Best solution	5.23E-06	0.021	0.0006	0.007	22.41	1.21E-05	0.0624
	Average solution	3.31E-04	1.41	0.0523	0.0126	29.41	6.12E-04	0.7723
chTLBO3	Best solution	1.32E-05	0.006	0.0005	0.0041	19.91	1.73E-06	0.0781
	Average solution	4.42E-04	1.333	0.0029	0.026	25.21	7.33E-04	0.8951
chTLBO4	Best solution	1.72E-08	0.012	0.0003	0.0081	23.91	8.72E-07	0.0318
	Average solution	8.12E-07	0.069	0.0051	0.0147	30.29	1.21E-07	0.1651
chTLBO5	Best solution	3.32E-06	0.010	0.0003	0.0027	22.32	1.53E-06	0.0482
	Average solution	9.32E-05	1.021	0.0052	0.0933	29.83	9.43E-05	0.4689
chTLBO6	Best solution	5.36E-06	0.005	0.0003	0.0059	22.37	9.06E-06	0.0762
	Average solution	1.43E-04	0.027	0.0742	0.0188	29.93	1.91E-05	0.1925
chTLBO7	Best solution	2.72E-06	0.009	0.0012	0.007	20.45	9.25E-06	0.0581
	Average solution	4.61E-05	0.066	0.0018	0.0591	22.21	1.32E-05	0.2342
chTLBO8	Best solution	2.12E-06	0.007	0.0002	0.0051	23.55	1.26E-06	0.0636
	Average solution	9.11E-05	0.011	0.0232	0.0141	28.51	7.88E-06	0.1891
chTLBO9	Best solution	7.56E-07	0.002	0.0002	0.003	17.4	2.51E-06	0.0752
	Average solution	1.99E-06	0.013	0.0113	0.0552	21.33	1.63E-06	0.1052
chTLBO10	Best solution	1.45E-06	0.010	0.0013	0.0021	18.62	1.42E-05	0.0761
	Average solution	7.32E-05	0.161	0.0184	0.0021	19.99	5.34E-05	0.1635

Abbreviation: TLBO, teaching learning-based optimization.

5.3 | Statistical results

Generally, Friedman test is a nonparametric statistical method which is useful for ranking the performance of the algorithms. The main objective of the Friedman test is to find whether any significant difference exists between the results of dissimilar algorithms. This is based on the null hypothesis that there is no difference in the performance of all algorithms.[47] The supreme algorithm becomes the lowest rank while the poorest accomplishment algorithm gets the highest rank as we can see in Figure 4.

6 | INTRUSION DETECTION SYSTEM

The identical and insignificant features existent in the data to cause a long-term difficult in the classification of traffic. It not only decrease the classification performance but also prevent a classifier for making correct decisions, entirely when substantial volumes of data are managed. The proposed method is used to find the most informative feature which can help to identify the important feature to classify attack correctly. The proposed algorithm uses support vector machine as a fitness function to select the extremely efficient features and to maximize the classification accuracy. The performance of the proposed method is evaluated using benchmark dataset of intrusion detection, including binary and multiclass problem with KDD Cup 99 dataset.

TABLE 5 Results of 10 chaotic maps and TLBO on benchmark functions

Methods	Performance	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}
TLBO	Best solution	−3542.54	30.12	0.0171	7.24E-05	6.64E-07	5.67E-06
	Average solution	−3450.61	33.49	0.0161	5.32E-06	4.67E-08	9.53E-07
chTLBO1	Best solution	−3831.15	25.88	0.0029	1.23E-08	3.81E-09	3.53E-08
	Average solution	−3800.61	26.55	0.071	0.24E-07	1.56E-08	4.88E-06
chTLBO2	Best solution	−3790.88	36.99	0.011	1.53E-07	2.65E-08	1.41E-06
	Average solution	−3661.25	37.87	0.024	1.91E-05	1.91E-06	1.69E-05
chTLBO3	Best solution	−2510.25	46.89	0.063	1.24E-07	6.12E-09	5.28E-05
	Average solution	−2464.43	48.51	2.012	1.91E-07	5.32E-08	9.06E-05
chTLBO4	Best solution	−2611.77	47.81	0.049	6.35E-07	3.32E-08	1.51E-06
	Average solution	−2013.23	47.99	1.031	8.87E-05	1.53E-09	5.33E-04
chTLBO5	Best solution	−2061.34	32.84	0.043	1.91E-07	2.88E-08	8.12E-07
	Average solution	−1897.52	35.69	2.054	1.58E-06	1.85E-07	8.80E-05
chTLBO6	Best solution	−2098.27	24.88	0.019	5.41E-08	5.36E-08	9.71E-07
	Average solution	−2053.32	32.35	0.001	8.81E-09	4.23E-07	1.35E-06
chTLBO7	Best solution	−2104.23	67.72	0.022	1.01E-07	2.80E-08	3.04E-07
	Average solution	−2132.45	68.54	0.008	4.65E-04	4.56E-06	8.05E-05
chTLBO8	Best solution	−2944.88	69.71	0.066	0.0123	8.98E-09	1.38E-07
	Average solution	−2864.23	69.96	0.007	0.3654	9.16E-06	1.65E-06
chTLBO9	Best solution	−2289.24	66.76	0.010	0.0089	6.27E-09	5.95E-07
	Average solution	−2212.65	69.99	0.0623	0.0461	1.43E-07	8.63E-06
chTLBO10	Best solution	−3412.12	35.82	0.0562	1.39E-08	7.91E-08	3.73E-07
	Average solution	−3233.89	34.84	0.0512	8.21E-07	1.61E-09	5.42E-06

Abbreviation: TLBO, teaching learning-based optimization.

6.1 | Intrusion detection based on proposed model

Multitypes attacks in network is required multi-SVM classifier to intrusion detection. In multiclass classification, one-against-one, one-against-all, and binary are the popular SVM method.[48] From Figure 5, binary SVM algorithm needs only (c – 1) or two-class SVM classifiers in the cases of c classes, while one-against-all SVM desires c two-class SVM classifiers where each one is trained with all the records and one-against-one SVM algorithm needs c(c – 1)/2 two class SVM classifiers where each one is trained on data from two classes. Understandably less two-class classifiers help accelerate the rate of training and modeling. Therefore, binary SVM classification algorithm is modified to construct detection model in this article.

The proposed method enriches the classification accuracy and also reduce the search complexity for generating the attribute sets concluded the high-dimensional IDS dataset. Generally, TLBO is widely applied to numerical optimization problems. In this article, we also intend a novel method as wrapper to solve the IDS problem. TLBO is population-based meta-heuristic search method inspired by teaching learning phenomena defined as a unit of cultural evolution that is capable of local modifications and to discriminate a type of an attack in the case of a new learner which helps to discover the early detection.[49] In addition, evaluate the performance of our method, we have used SVM as fitness function and find the best learner with optimal number features. The overall process of proposed algorithm for feature selection is defined:

- Make the randomly learners for proposed. Produce population (nPop) is set as 50. Each population consists number of attributes selected in filter.

TABLE 6 Results of 10 chaotic maps and TLBO on benchmark functions

Methods	Performance	F_{14}	F_{15}	F_{16}	F_{17}	F_{18}	F_{19}	F_{20}	F_{21}	F_{22}	F_{23}
TLBO	Best solution	3.853	0.0132	−1.0894	0.3982	3	−3.681	−1.352	−5.044	−11.89	−10.55
	Average solution	4.161	0.0094	−0.5614	0.3857	2.889	−3.156	−1.271	−4.749	−10.40	−9.32
chTLBO1	Best solution	2.051	0.0003	−1.0352	0.3888	3	−3.694	−1.211	−5.615	−11.32	−10.61
	Average solution	3.236	0.0008	−0.5874	0.3817	2.999	−3.151	−1.267	−4.862	−10.41	−9.24
chTLBO2	Best solution	3.142	0.0004	−1.0321	0.3981	3	−3.619	−1.289	−5.041	−11.86	−5.54
	Average solution	3.834	0.0006	−0.5674	0.3987	2.991	−3.426	−1.656	−4.872	−10.48	−4.89
chTLBO3	Best solution	2.457	0.0026	−1.0311	0.3976	3	−3.248	−1.628	−5.056	−11.97	−5.12
	Average solution	2.986	0.0052	−0.5714	0.3987	2.998	−3.844	−1.687	−4.852	−10.47	−4.75
chTLBO4	Best solution	3.968	0.0025	−1.0312	0.3978	3	−2.488	−1.233	−5.055	−11.98	−5.12
	Average solution	3.754	1.377	−0.5674	0.3877	2.989	−1.996	−1.866	−4.867	−10.42	−4.79
chTLBO5	Best solution	2.121	0.006	−1.0344	0.3958	3	−3.179	−2.017	−5.053	−11.93	−5.17
	Average solution	1.884	0.0049	−0.5674	0.3987	2.992	−3.009	−2.104	−4.861	−10.44	−4.95
chTLBO6	Best solution	2.081	0.0077	−1.0313	0.3976	3	−3.578	−1.316	−5.054	−11.97	−10.54
	Average solution	2.355	0.0677	−0.5674	0.3987	2.993	−3.520	−1.990	−4.859	−10.47	−9.24
chTLBO7	Best solution	3.129	0.0006	−1.0311	0.3972	3	−3.210	−1.14	−1.879	−11.93	−10.52
	Average solution	3.011	0.0689	−0.5685	0.3978	2.999	−3.076	−1.553	−0.003	−10.43	−9.23
chTLBO8	Best solution	3.922	0.0083	−1.0312	0.3978	3	−3.388	−1.065	−1.881	−11.98	−10.53
	Average solution	2.895	0.0786	−0.5674	0.3977	2.999	−3.185	−1.897	−0.003	−10.44	−9.23
chTLBO9	Best solution	11.042	0.0113	−1.0312	0.3978	3	−2.951	−1.05	−3.904	−11.98	−10.54
	Average solution	8.558	0.0456	−0.5662	0.3981	2.999	−2.851	1.883	−3.159	−10.46	−9.24
chTLBO10	Best solution	9.998	0.0091	−1.0312	0.3978	3	−3.688	−1.565	−6.855	−11.98	−10.53
	Average solution	10.241	0.0361	−0.5674	0.3987	2.999	−3.608	−1.977	−6.054	−10.42	−9.23

Abbreviation: TLBO, teaching learning-based optimization.

- Apply binary encoding scheme for initialization of population and on SVM parameters. After encoding, each individual length can be few features present in the reduced dataset.
- For each learner, find out the fitness function using SVM classifier considering classification accuracy as the fitness function.
- Select a highly scored value of individual as consider a teacher in teacher phase.
- Then use the learner phase to make a new updated individual.
- According to the new value of individual update the old individual.
- Obtain the parameters of SVM model. If, the recent best fitness value meets the termination condition, if yes, go to next; otherwise, go to previous step.
- Output as an optimal subset of attributes.

6.2 | Experimental results and discussion

At this time, there are only a few public datasets are available for intrusion detection assessment. Among these datasets, KDD Cup 99 conglomerate has been used in the works to assess the IDS performance and to identify the type of attacks.[50] Each record has 41 attributes represents different

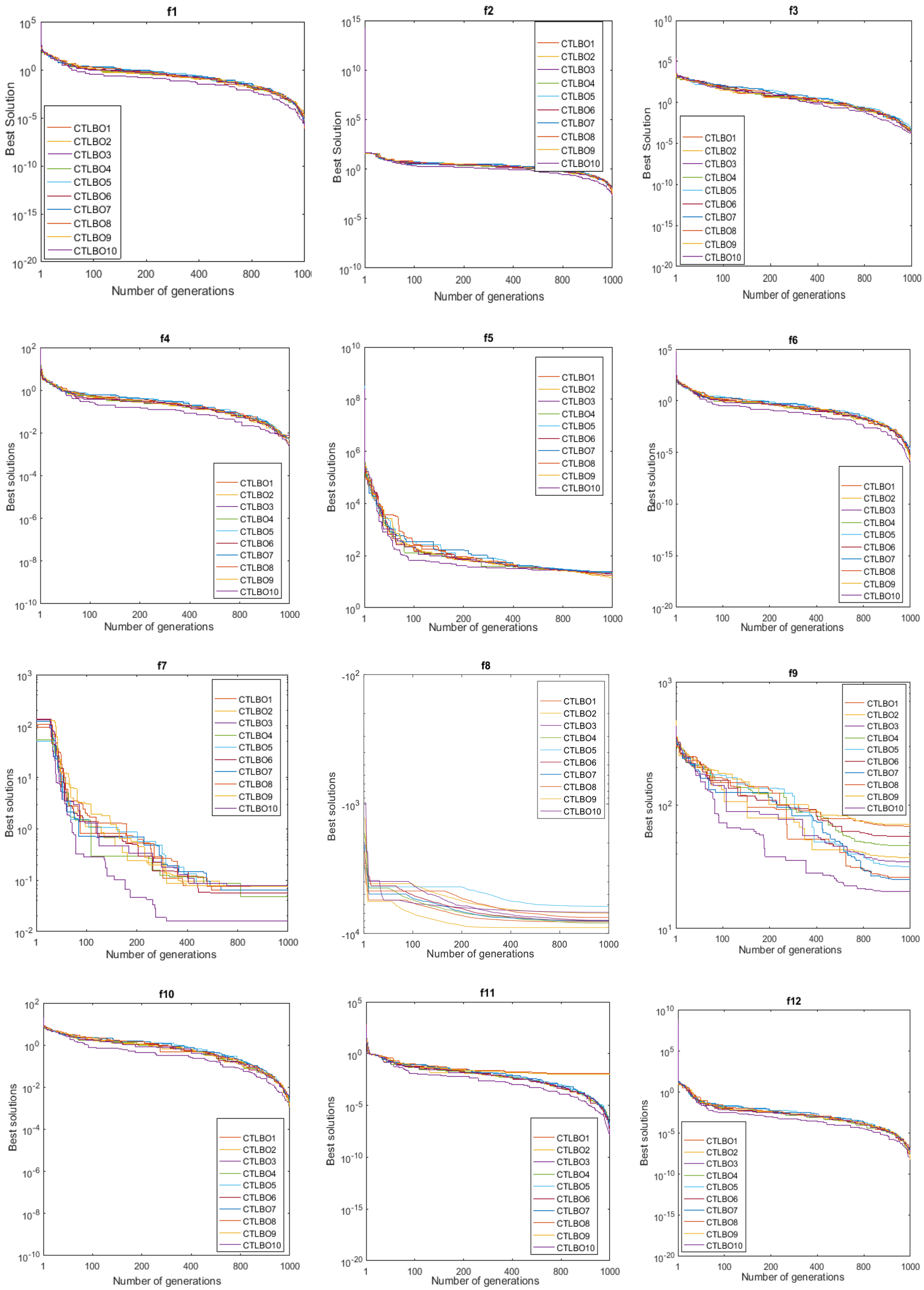

FIGURE 3 Convergence graph on 23 benchmark functions using teaching learning-based optimization variants method

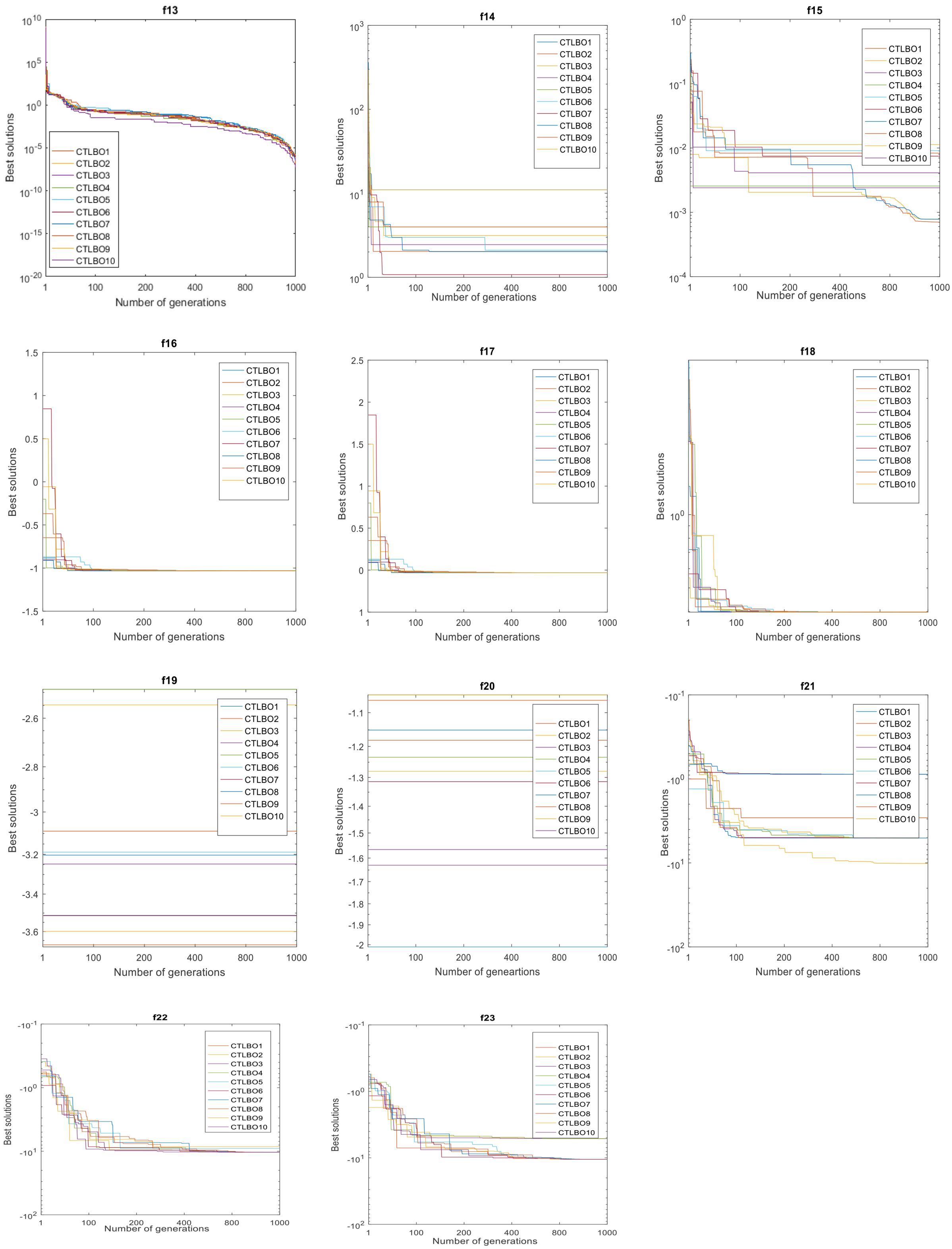

FIGURE 3 (Continued)

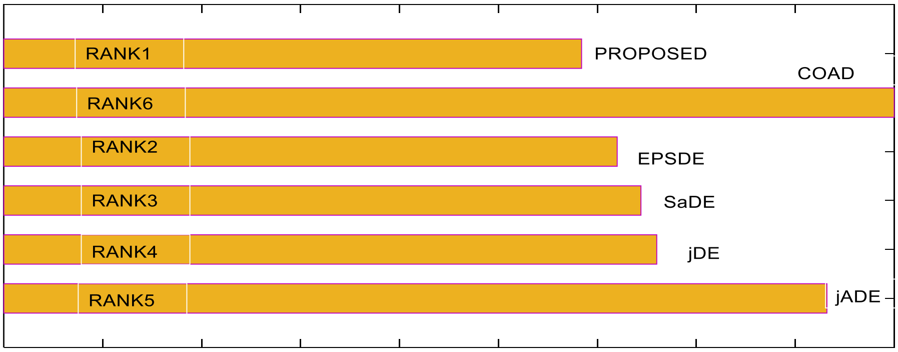

FIGURE 4 Ranking of existing algorithms and proposed method by Friedman test

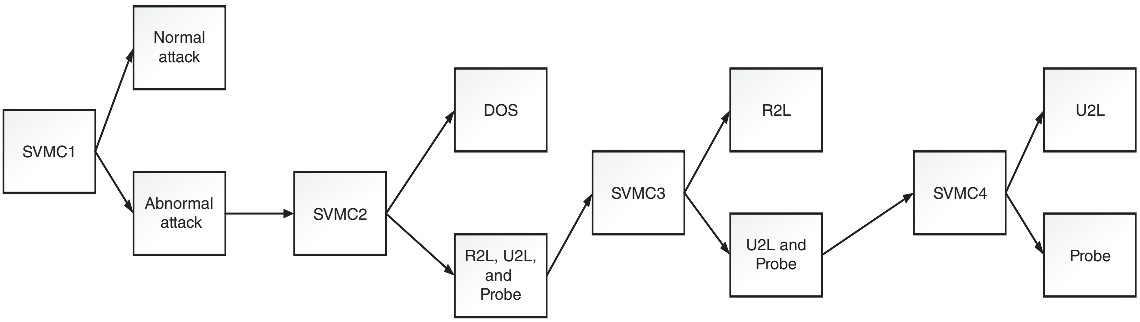

FIGURE 5 The scheme of intrusion detection based on improved SVM model

features types, namely, the feature name, records, and feature portrayal. According to Reference [51], the majority of the IDS experiments were performed on the KDD Cup 99 datasets. In addition, these datasets have different data values and several feature numbers that meet exhaustive tests to validate feature selection methods.

The KDD Cup 99 dataset features have dissimilar data types. Besides from normal data, records that relate to the 39 different attack types are found in the KDD Cup 99 dataset. All of these attack can be considered into four different type of attacks such as normal attack, DoS attack, Probe attack, R2L attack, and U2R attack. In more details, denial of service (DoS) has 10 types of attack, Probing (Probe) has six types of attack, unauthorized access from a remote machine (R2L) has 16 types of attack, and unauthorized access to the local to user (U2R) has seven types of attack.

6.3 | Comparison of proposed method with exiting state-of-arts

In this experiment, we performed cross-validation 10-fold using three classifier in the IDS dataset as seen in Table 7 in terms of classification performance. The results of this experiment are shown in Table 7 as accuracy, precision, and *F*-measure (in percentage). As we can see, the accuracy of the classification is not very interesting, especially for the k-NN classifier. The higher values of these criteria represent an excellent classification performance. It demonstrates that IDS combined with chaos achieved an accuracy of 97.25% in KDD Cup 99 with SVM, and significantly outperform to remaining classification methods.

6.4 | Comparative study between nature-inspired algorithms in fold-wise

To show the performance of proposed, experimental have been done to make examinations with all type of attacks. As stated in above-section, the KDD Cup 99 dataset is partitioned into five distinct labels such as normal, DoS, Probe, U2R, and R2L attacks, respectively. Table 8 demonstrates the optimal six attributes chosen for the distinctive attack labels on the KDD Cup 99 dataset. The outcomes outlined in these tables obviously prove that the proposed algorithm display likely results contrasted with different models.

From the Table 9, we observed that five optimal attributes have been selected by the proposed method which can be identify the attacks in the networks for intrusion detection. These attributes plays a significant role in IDS system. Table 9 shows the optimal selected features and also gives a short description of attributes, namely, 4, 13, 21, 27, and 38.

TABLE 7 Percentage of average performance in four wrapper method in all attacks

Dataset	Measure	GA	PSO	TLBO	Proposed
SVM	Accuracy	89.85	90.88	91.62	97.25
	Precision	89.03	88.63	90.63	96.13
	F-measure	88.97	88.52	90.47	94.52
NB	Accuracy	85.15	86.02	90.15	95.15
	Precision	83.09	85.93	88.36	94.63
	F-measure	82.62	84.78	87.12	93.29
k-NN	Accuracy	84.01	85.01	87.36	92.65
	Precision	83.15	84.97	85.67	90.12
	F-measure	83.06	83.98	86.13	91.63

Abbreviations: GA, genetic algorithm; PSO, particle swarm optimization; TLBO, teaching learning-based optimization.

TABLE 8 Comparison the experimental performance in all attack

Methods	Measures	Fold1	Fold2	Fold3	Fold4	Fold5	Fold6	Fold7	Fold8	Fold9	Fold10
ChTLBO-SVM	DR (%)	95.13	95.22	95.07	96.17	96.86	96.07	97.11	97.15	98.02	96.99
	FAR (%)	1.123	1.037	0.954	1.520	0.998	1.021	1.000	1.000	0.998	0.976
	EtrD (min)	2.54	3.53	4.62	5.12	3.12	4.52	2.99	3.23	3.52	2.35
	EteD (min)	3.54	4.03	5.32	6.98	5.07	6.38	4.23	4.35	5.03	5.01
TLBO-SVM	DR (%)	92.62	92.89	91.34	91.35	92.13	92.53	94.13	94.12	93.11	90.32
	FAR (%)	2.236	1.956	1.998	3.023	2.974	2.658	3.847	2.036	2.036	1.689
	EtrD (min)	3.24	5.34	5.32	4.89	5.23	5.63	5.23	6.53	7.01	5.12
	EteD (min)	5.01	6.21	7.34	5.23	7.52	7.03	6.34	9.23	8.23	7.23
GA-SVM	DR (%)	89.56	88.35	90.32	88.64	89.64	89.13	89.37	88.35	90.11	89.26
	FAR (%)	6.265	3.265	4.253	5.032	6.007	6.089	4.023	4.325	4.032	4.320
	EtrD (min)	4.89	6.03	7.12	5.23	6.12	6.03	7.81	7.96	8.23	8.52
	EteD (min)	5.23	7.52	8.37	6.37	7.99	8.23	8.97	8.38	10.28	11.23
PSO-SVM	DR (%)	90.37	88.63	89.15	89.37	89.17	89.62	90.13	89.63	90.11	89.17
	FAR (%)	6.32	11.23	14.23	10.12	15.23	12.32	17.32	11.64	10.35	12.36
	EtrD (min)	15.56	18.53	16.42	18.16	20.01	17.89	19.34	17.52	21.52	19.03
	EteD (min)	19.61	19.87	21.03	22.65	25.95	19.62	21.78	20.36	25.32	26.35

Abbreviations: GA, genetic algorithm; PSO, particle swarm optimization; TLBO, teaching learning-based optimization.

TABLE 9 Description of selected features

Selected feature	Feature description
4	Flag: Status of the connection—Normal or Error
13	num_compromised: Number of compromised conditions
21	is_host_login: 1 if the login belongs to the hot list, that is, root or admin else 0
27	rerror_rate: The percentage of connections that have activated the flag (4) REJ, among the connections aggregated in the count (23)
38	dst_host_serror_rate: The percentage of connections that have activated the flag (4) s0, s1, s2, or s3, among the connections aggregated in dst_host_count (32)

7 | CONCLUSION

The balance between exploitation and exploration is guaranteed for finding global optimum, especially for multimodal functions. Too much exploration can slow the convergence rate and discarded the computation efforts, however exploration is significant for accessing the global optimum. Besides, too much exploitation tends to trap the algorithm into local optimum. Keeping this in mind, we have investigated a new intelligent algorithm by improvisation of original TLBO which is applied to solve global optimization problems. Unlike original TLBO, proposed algorithm tried to make a good balance. In proposed method, we have integrated chaos maps in original TLBO to produce good solutions and to enhance the capability of TLBO. By comparing different chaotic TLBO, algorithm that practices the logistic map for providing the values is the best chaotic TLBO. In the experiments study, by analyzing the properties of the existing unimodal and multimodal test functions. Generally speaking, the results proved that 10 different chaotic maps significantly increase the performance of original TLBO. These variants are testified on benchmark test suites, and their performances are compared with each other. Based on the results of the 10 chaos TLBO variants on the selected test suites, we concluded that chTLBO1 significantly improves the performance regarding convergence speed, stability, and robustness and gives the best performance compared with other nine chaos TLBO variants on multimodal problems. Hence, all variants of TLBO algorithm may be an excellent alternative to deal with complex numerical optimization problems. In particular intrusion detection, proposed method has achieved up to 97.25% classification accuracy with optimal feature sets for KDD Cup 99 dataset. For future work, it would be interesting to work proposed technique to resolve the image segmentation, computer vision, and bioinformatics based problems.

DATA AVAILABILITY STATEMENT

Data openly available in a public repository (UCI), URL: https://kdd.ics.uci.edu/databases/kddcup99/, Reference [50].

ORCID

Alok Kumar Shukla https://orcid.org/0000-0003-4352-4018

REFERENCES

1. Arunkumar N, Mohammed MA, Mostafa SA, Ibrahim DA, Rodrigues JJ, de Albuquerque VH. Fully automatic model-based segmentation and classification approach for MRI brain tumor using artificial neural networks. *Concurr Comput Pract Exper*. 2018;32(May):1-9. https://doi.org/10.1002/cpe.4962
2. Liu T, Jiao L, Ma W, Shang R. Quantum-behaved particle swarm optimization with collaborative attractors for nonlinear numerical problems. *Commun Nonlinear Sci Numer Simul*. 2017;44:167-183. https://doi.org/10.1016/j.cnsns.2016.08.001
3. Shukla AK, Singh P, Vardhan M. Hybrid TLBO-GSA strategy for constrained and unconstrained engineering optimization functions. *Hybrid Metaheuristics: Research and Applications*. 6 edition. World Scientific; 2021;60:6013–6033.
4. Cao Z, Wang L, Hei X. A global-best guided phase based optimization algorithm for scalable optimization problems and its application. *J Comput Sci*. 2018;25:38-49. https://doi.org/10.1016/j.jocs.2018.02.001
5. Prayogo MCD. Fuzzy adaptive teaching – learning-based optimization for global numerical optimization. *Neural Comput & Applic*. 2018;29(2):309-327. https://doi.org/10.1007/s00521-016-2449-7
6. Mesejo P, Ugolotti R, Di Cunto F, Giacobini M, Cagnoni S. Automatic hippocampus localization in histological images using differential evolution-based deformable models. *Pattern Recogn Lett*. 2013;34(3):299-307. https://doi.org/10.1016/j.patrec.2012.10.012
7. Burke EK, Gendreau M, Hyde M, et al. Hyper-heuristics: a survey of the state of the art. *J Oper Res Soc*. 2013;64(12):1695-1724. https://doi.org/10.1057/jors.2013.71
8. Arora S, Anand P. Chaotic grasshopper optimization algorithm for global optimization. *Neural Comput Appl*. 2019;31(8):4385–4405. https://doi.org/10.1007/s00521-018-3343-2
9. Torabi S, Sa F. Improved raven roosting optimization algorithm (IRRO). *Swarm Evol Comput*. 2018;40:144-154. https://doi.org/10.1016/j.swevo.2017.11.006
10. Domínguez-isidro S, Mezura-montes E. A cost-benefit local search coordination in multimeme differential evolution for constrained numerical optimization problems. *Swarm Evol Comput*. 2018;39:249-266. https://doi.org/10.1016/j.swevo.2017.10.006
11. Thakur M. A new genetic algorithm for global optimization of multimodal continuous functions. *J Comput Sci*. 2013;5(2):298-311. https://doi.org/10.1016/j.jocs.2013.05.005
12. Shukla AK. Detection of anomaly intrusion utilizing self-adaptive grasshopper optimization algorithm. *Neural Comput Appl*. 2020;24:1–21. https://doi.org/10.1007/s00521-020-05500-7
13. Ghosh S, Das S, Kundu D. An inertia-adaptive particle swarm system with particle mobility factor for improved global optimization. *Neural Comput Appl*. 2012;21:237-250. https://doi.org/10.1007/s00521-010-0356-x
14. Yang D, Li G, Cheng G. On the efficiency of chaos optimization algorithms for global optimization. *Chaos Soliton Fract*. 2007;34:1366-1375. https://doi.org/10.1016/j.chaos.2006.04.057
15. Lewis A. Biogeography-based optimisation with chaos. *Neural Comput Appl*. 2014;25:1077-1097. https://doi.org/10.1007/s00521-014-1597-x
16. Shi Y, Eberhart RC. Fuzzy adaptive particle swarm optimization. In: Proceedings of the 2001 congress on evolutionary computation (IEEE); Vol. 1, 2001:101-106; IEEE.
17. Rabiee A, Jalilzadeh S, Nojavan S. Probabilistic multi objective optimal reactive power dispatch considering load uncertainties using Monte Carlo simulations. *J Oper Autom Power Eng*. 2015;3(1):83-93.

18. Güçyetmez M, Çam E. A new hybrid algorithm with genetic-teaching learning optimization (G-TLBO) technique for optimizing of power flow in wind-thermal power systems. *Electr Eng*. 2016;98(2):145-157. https://doi.org/10.1007/s00202-015-0357-y
19. Wang X, Yu K, Wang X, Wang Z. An improved teaching-learning-based optimization algorithm for numerical and engineering optimization problems. *J Intell Manuf*. 2014;27(4):831-843. https://doi.org/10.1007/s10845-014-0918-3
20. Rao RV, Savsani VJ, Vakharia DP. Teaching learning-based optimization: a novel method for constrained mechanical design optimization problems. *Comput Des*. 2011;43(3):303-315. https://doi.org/10.1016/j.cad.2010.12.015
21. Zhang Y, Jin Z, Chen Y. Hybrid teaching – learning-based optimization and neural network algorithm for engineering design optimization problems. *Knowl Based Syst*. 2020;187:104836. https://doi.org/10.1016/j.knosys.2019.07.007
22. Satapathy SC, Naik A. Modified teaching–learning-based optimization algorithm for global numerical optimization—a comparative study. *Swarm Evol Comput*. 2014;16:28-37. https://doi.org/10.1016/j.swevo.2013.12.005
23. Shukla AK, Singh P, Vardhan M. Neighbour teaching learning based optimization for global optimization problems. *J Intell Fuzzy Syst*. 2018;34(3):1583-1594. https://doi.org/10.3233/JIFS-169453
24. Birashk A, Kazemi Kordestani J, Meybodi MR. Cellular teaching-learning-based optimization approach for dynamic multi-objective problems. *Knowl Based Syst*. 2018;141:148-177. https://doi.org/10.1016/j.knosys.2017.11.016
25. Taher Niknam MSS, Golestaneh F. θ-Multiobjective teaching–learning-based optimization for dynamic economic emission dispatch. *IEEE Syst J*. 2012;6(2):341-352.
26. Murty MR, Naik A, Murthy JVR, Reddy PVGDP. Automatic clustering using teaching learning based optimization. *Appl Math*. 2014;5(8):1202-1211.
27. Herwan M, Mustaffa Z, Mawardi M, Daniyal H. Barnacles mating optimizer: a new bio-inspired algorithm for solving engineering optimization problems ☆. *Eng Appl Artif Intel*. 2020;87:103330. https://doi.org/10.1016/j.engappai.2019.103330
28. Khosravi A, Malekan M, Pabon JJG, Zhao X, Assad MEH. Design parameter modeling of solar power tower system using adaptive neuro-fuzzy inference system optimized with a combination of genetic algorithm and teaching learning-based optimization algorithm. *J Clean Prod*. 2019;244:118904. https://doi.org/10.1016/j.jclepro.2019.118904
29. Aich U, Banerjee S. Application of teaching learning based optimization procedure for the development of SVM learned EDM process and its pseudo Pareto optimization. *Appl Soft Comput J*. 2016;39:64-83. https://doi.org/10.1016/j.asoc.2015.11.002
30. Shahbeig S, Sadegh M, Rahideh A. A fuzzy multi-objective hybrid TLBO – PSO approach to select the associated genes with breast cancer. *Signal Process*. 2017;131:58-65. https://doi.org/10.1016/j.sigpro.2016.07.035
31. Kumar Y, Singh PK. A chaotic teaching learning based optimization algorithm for clustering problems. *Appl. Intell*. 2018;49(3):1036–1062.
32. Manzoor I, Kumar N. A feature reduced intrusion detection system using ANN classifier. *Expert Syst Appl*. 2017;88:249-257. https://doi.org/10.1016/j.eswa.2017.07.005
33. Liu Y, Shi Y, Chen H, et al. Chaos-assisted multi-population salp swarm algorithms: framework and case studies. *Expert Syst Appl*. 2021;168(October):114369. https://doi.org/10.1016/j.eswa.2020.114369
34. Jiang J, Yang X, Meng X, Li K. Enhance chaotic gravitational search algorithm (CGSA) by balance adjustment mechanism and sine randomness function for continuous optimization problems. *Physica A*. 2020;537:122621. https://doi.org/10.1016/j.physa.2019.122621
35. Aydin I, Karakose M, Akin E. Chaotic-based hybrid negative selection algorithm and its applications in fault and anomaly detection. *Expert Syst Appl*. 2010;37(7):5285-5294. https://doi.org/10.1016/j.eswa.2010.01.011
36. De E, De E, Ortiz A, Ortega J, Martínez-álvarez A. Feature selection by multi-objective optimisation: application to network anomaly detection by hierarchical self-organising maps. *Knowl Based Syst*. 2014;71:322-338. https://doi.org/10.1016/j.knosys.2014.08.013
37. Song J. Statistical analysis of honeypot data and building of Kyoto 2006 + dataset for NIDS evaluation. Paper presented at: Proceedings of the 1st Workshop on Building Analysis Datasets and Gathering Experience Returns for Security; 2011:29-36.
38. Basseur M, Goëffon A. Climbing combinatorial fitness landscapes. *Appl Soft Comput J*. 2015;30:688-704. https://doi.org/10.1016/j.asoc.2015.01.047
39. Karaboga D, Basturk B. On the performance of artificial bee colony (ABC) algorithm. *Appl Soft Comput*. 2008;8:687-697. https://doi.org/10.1016/j.asoc.2007.05.007
40. Raja BD, Jhala RL, Patel V. Multi-objective optimization of a rotary regenerator using tutorial training and self-learning inspired teaching-learning based optimization algorithm (TS-TLBO). *Appl Therm Eng*. 2016;93:456-467. https://doi.org/10.1016/j.applthermaleng.2015.10.013
41. Rizk-allah RM, Hassanien AE, Bhattacharyya S. Chaotic crow search algorithm for fractional optimization problems. *Appl Soft Comput J*. 2018;71:1161–1175. https://doi.org/10.1016/j.asoc.2018.03.019
42. Patel SJ, Toshniwal A, Kheraj V. A novel teaching-learning based optimization approach for design of broad-band anti-reflection coatings. *Swarm Evol Comput*. 2016;34:1-7. https://doi.org/10.1016/j.swevo.2016.12.003
43. Farah A, Guesmi T, Hadj Abdallah H, Ouali A. A novel chaotic teaching-learning-based optimization algorithm for multi-machine power system stabilizers design problem. *Int J Electr Power Energy Syst*. 2016;77:197-209. https://doi.org/10.1016/j.ijepes.2015.11.050
44. Mirjalili S, Hashim SZM. A new hybrid PSOGSA algorithm for function optimization. Paper presented at: Proceedings of ICCIA 2010–2010 International Conference on Computer and Information Application; 2010:374-377. https://doi.org/10.1109/ICCIA.2010.6141614.
45. Suganthan PN, Hansen N, Liang JJ, Deb K, Chen Y-P, Auger A, Tiwari S. Problem definitions and evaluation criteria for the CEC 2006 special session on constrained real-parameter optimization. *KanGAL*. 2006;41(8):8-31.
46. Suganthan PN, Hansen N, Liang JJ, et al. *Problem Definitions and Evaluation Criteria for the CEC 2005 Special Session on Real-Parameter Optimization Technical Report KanGAL#2005*. Kanpur, India: Nanyang Techical. University, Singapore, IIT Kanpur; 2005.
47. SreeRanjinik S, Murugan S. Memory based hybrid dragonfly algorithm for numerical optimization problems. *Expert Syst Appl*. 2017;83:63-78. https://doi.org/10.1016/j.eswa.2017.04.033
48. Aladeemy M, Tutun S, Khasawneh MT. A new hybrid approach for feature selection and support vector machine model selection based on self-adaptive cohort intelligence. *Expert Syst Appl*. 2017;88:118-131. https://doi.org/10.1016/j.eswa.2017.06.030
49. Soufan O, Kleftogiannis D, Kalnis P, Bajic VB. DWFS: a wrapper feature selection tool based on a parallel genetic algorithm. *PLoS One*. 2015;10(2):e0117988. https://doi.org/10.1371/journal.pone.0117988
50. Tavallaee M, Bagheri E, Lu W, Ghorbani AA. A detailed analysis of the KDD CUP 99 data set. Paper presented at: Proceedings of the 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications; 2009:1-6.

51. Buczak A, Guven E. A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Commun Surv Tutor*. 2016;18(2):1153-1175. https://doi.org/10.1109/COMST.2015.2494502

How to cite this article: Shukla AK. Chaos teaching learning based algorithm for large-scale global optimization problem and its application. *Concurrency Computat Pract Exper*. 2022;34(1):e6514. https://doi.org/10.1002/cpe.6514