การหาค่าเหมาะสมโดยขั้นตอนวิธีซาตินโบเวอร์เบิร์ดแบบอลวน สำหรับการเพิ่มประสิทธิภาพของการหาค่าเหมาะสมของฟังก์ชันเชิงตัวเลข

Chaotic satin bowerbird optimization for improving the efficiency of numerical function optimization

ธนัชพงษ์ วังคำหาญ[1*], อนงค์นาถ โรจนกร วังคำหาญ[1]
Tanachapong Wangkhamhan[1*], Anongnart Rotjanakorn Wangkhamhan[1]

Received: 25 December 2019 ; Revised: 31 May 2020 ; Accepted: 25 August 2020

บทคัดย่อ

การเพิ่มประสิทธิภาพซาตินโบเวอร์เบิร์ด เป็นขั้นตอนวิธีเมตาฮิวริสติกที่พัฒนาเมื่อเร็วๆ นี้ ปัญหาหลักที่ขั้นตอนวิธีซาตินโบเวอร์เบิร์ดเผชิญอยู่ซึ่งได้รับการพิสูจน์แล้วอย่างชัดเจน การติดอยู่ในค่าคำตอบดีที่สุดเฉพาะที่อย่างง่ายดาย มีความแม่นยำต่ำและความเร็วในลู่เข้าแก้ปัญหาการหาค่าเหมาะสมที่ช้า ดังนั้นในความพยายามที่จะเพิ่มความเร็วในการลู่เข้าแก้ปัญหาการหาค่าเหมาะสมที่แท้จริง และได้รับประสิทธิภาพที่ดีขึ้น บทความนี้จะนำเสนอทฤษฎีความอลวนในกระบวนการเพิ่มประสิทธิภาพขั้นตอนวิธีซาตินโบเวอร์เบิร์ด ตัวแปรความวุ่นวายในแมปจะถูกนำมาพิจารณาโดยการนำเสนอวิธีความอลวนกับขั้นตอนวิธีซาตินโบเวอร์เบิร์ด เพื่อที่จะแทนที่ตัวแปรหลัก (α) ซึ่งช่วยในการควบคุมทั้งการสำรวจพื้นที่และการนำไปใช้ประโยชน์ของขั้นตอนวิธีซาตินโบเวอร์เบิร์ด เราทดสอบความอลวนกับขั้นตอนวิธีซาตินโบเวอร์เบิร์ด ผ่านการทดสอบการเพิ่มประสิทธิภาพฟังก์ชันตัวเลข ผลลัพธ์เชิงตัวเลขระบุว่าขั้นตอนวิธีที่เราได้นำเสนอนั้นมีประสิทธิภาพเหนือกว่าขั้นตอนวิธีการเพิ่มประสิทธิภาพอื่นทั้ง 11 ขั้นตอนวิธี

คำสำคัญ: การเพิ่มประสิทธิภาพซาตินโบเวอร์เบิร์ด ขั้นตอนวิธีเมตาฮิวริสติก ทฤษฎีความอลวน แมปความอลวน

Abstract

The Satin Bowerbird Optimization (SBO) is a recently developed meta-heuristic optimization algorithm. The main problem faced by the SBO is that it has been empirically demonstrated to become easily trapped into local optimal solutions, creating low precision and slow convergence speeds. Therefore, in an effort to enhance global convergence speeds, and to obtain better performance, this paper introduces Chaos Theory into the SBO optimization process. Various chaotic maps were considered in the proposed Chaotic-SBO (CSBO) method in order to replace the main parameter's greatest step size (α), which assists in controlling both exploration and exploitation. We tested CSBO algorithms through experiments with the numerical function optimization. The numerical results indicate that the CSBO algorithm outperformed 11 other optimization algorithms.

Keywords: Satin Bowerbird Optimization, Meta-heuristic algorithm, chaos theory, chaotic map.

[1] อาจารย์, คณะวิทยาศาสตร์และเทคโนโลยีสุขภาพ มหาวิทยาลัยกาฬสินธุ์ อำเภอเมืองกาฬสินธุ์ จังหวัดกาฬสินธุ์ 46000
[1] Lecturer, Faculty of Science and Health Technology, Kalasin University, Kalasin 46000, Thailand.
* Corresponding author ; Tanachapong Wangkhamhan, Faculty of Management Science, Kalasin University, Kalasin 46000, Thailand.
tanachapong.w@hotmail.com.

Introduction

The goal of the optimization problem is to search for a maximum or minimum of an objective function value in widely varied local optima, under highly complex constraints, and in a reasonable amount of time (Yang *et al*., 2014). Consequently, chaotic sequences generated by means of chaotic maps have been used in the development of global optimization techniques. The first introduction of chaos into the optimization challenge was the Chaos Optimization Algorithm (COA) in 1963, by E.N. Lorenz (Lorenz, 1963). The COA represents the bounded, unstable, dynamic behavior that exhibits sensitive dependence on its initial conditions (Yuan *et al*., 2014) named chaos optimization algorithm (COA. Uniquely characteristic of chaotic behavior, the COA carries out global exploration searches at higher speeds than stochastic ergodic searches, which are dependent on probabilities (Yuan *et al*., 2015) all individuals in the PCOA search independently without utilizing the fitness and diversity information of the population. In view of the limitation of PCOA, a novel PCOA with migration and merging operation (denoted as MMO-PCOA.

Chaos is a characteristic of several nonlinear systems as motion distributes within a specific range, as it possesses degrees of uncertainty, ergodicity, and stochasticity. Many researchers therefore use the characteristics of chaotic ergodicity to solve for the global optimal solution of complex nonlinear multi-peak problems, by weakening the randomness or constant parameters of the metaheuristic optimization algorithm (Huang *et al*., 2015) which is widely used to solve many optimization problems. However, it has been empirically demonstrated to easily get trapped into local optimal solutions and cause low precision. Therefore, in this work, we propose five modified Chaos-enhanced Cuckoo search (CCS. As a result, most current work is devoted to the improvement of global optimization algorithms to tackle the abovementioned shortcomings. Further interest has been developed in the field of hybrid algorithms, especially in typical and emerging heuristic optimization algorithms; such as the migration and merging operation (Yuan *et al*., 2015) all individuals in the PCOA search independently without utilizing the fitness and diversity information of the population. In view of the limitation of PCOA, a novel PCOA with migration and merging operation (denoted as MMO-PCOA, cuckoo search optimization algorithm (Huang *et al*., 2015)which is widely used to solve many optimization problems. However, it has been empirically demonstrated to easily get trapped into local optimal solutions and cause low precision. Therefore, in this work, we propose five modified Chaos-enhanced Cuckoo search (CCS, firefly algorithm (Gandomi *et al*., 2013), gravitational search algorithm (Mirjalili and Gandomi, 2017), whale optimization algorithm (Kaur & Arora, 2018), crow search algorithm (Problems *et al*., 2018), league championship algorithm (Wangchamhan *et al*., 2017)but the produced solution does not produce optimum clusters. This paper proposes three algorithms (i, salp swarm algorithm (Sayed *et al*., 2018), and the krill herd algorithm (Wang *et al*., 2014)Gandomi and Alavi proposed a meta-heuristic optimization algorithm, called Krill Herd (KH; all of which were hybridized with the COA. Various simulation results and applications in each of these references have proven the solution diversity and global optimization capacity of each chaos-based optimization algorithm.

The standard Satin Bowerbird Optimizer (SBO) was first proposed by S. H. Samareh Moosavi and V. Khatibi Bardsiri (Moosavi and Bardsiri, 2017)development effort estimation has become a challenging issue which must be seriously considered at the early stages of project. Insufficient information and uncertain requirements are the main reasons behind unreliable estimations in this area. Although numerous effort estimation models have been proposed during the last decade, accuracy level is not satisfying enough. This paper presents a new model based on a combination of adaptive neuro-fuzzy inference system (ANFIS in 2017, to optimize adaptive neuro-fuzzy inference system (ANFIS) for the purpose of effort estimation of software development. Its algorithm was bio-inspired by Satin Bowerbirds living in the rainforests and mesic habitats of Australia. Through the breeding principle of male-attracting-female, the male bowerbird attracts the female with the construction of a specialized bower. This technique, which is population-based on a stochastic optimization algorithm (Chintam and Daniel, 2018); is very robust, straightforward, and efficient. Details of the original SBO and the literature related to its applications are presented in Sections 4 and 5.

The principle concern we faced in our research was the way to introduce the Chaotic Satin Bowerbird Optimizer (CSBO) based methods in which different chaotic systems are used to replace the critical parameters of the SBO. Through this method, we intended to enhance the global searching ability of the SBO and increase its ability to stick on a local solution. The simulation results demonstrated the improved performance of the CSBO with the application of the deterministic chaotic signals, as opposed to the constant parameters of the SBO.

The remainder of this paper is organized in six sections. Section 2 briefly describes the SBO algorithm ; Section 3 describes the chaotic maps for the SBO ; the proposed CSBO approach is detailed in Section 4 ; and comparisons of the CSBO with other optimization algorithms are found in Section 5. Our conclusions and future scope of our research are presented in Section 6.

The original SBO algorithm

The SBO algorithm starts by creating a population of random uniform distribution, through the consideration of both the lower and upper limit parameters. After that, each position is defined as a dimensional vector of the parameters, which must be optimized. The probability of such defines the attractiveness of the bower. A female satin bower bird selects a bower (nest) based on its probability and is able to calculate the probability of each population member through Eqs. (1) and (2), below.

$$\mathrm{Prob}_i = \frac{\mathrm{fit}_i}{\sum_{n=1}^{\mathrm{NB}} \mathrm{fit}_n}, \tag{1}$$

$$\mathrm{fit}_i \begin{cases} \dfrac{1}{1+f(x_i)}, f(x_i) \geq 0 \\ 1+\left|f(x_i)\right|, f(x_i) < 0 \end{cases}, \tag{2}$$

where NB is the population size of the bower, is the fitness value of the i^{th} solution, and $f(x_i)$ is the fitness value of i^{th} bower. To find the position of the best bower, the SBO algorithm utilizes the concept of *elitism*, which allows the best solution to be preserved at each stage of the optimization process. The SBO algorithm replicates the concept of birds building their nests using their natural instincts. In the mating season, the male satin bower bird uses his natural instincts to build and decorate his bower, in an attempt to attract female birds. We may infer that the male birds rely upon their experience to influence their creative decisions in building their bower ; therefore, more experienced birds will build more attractive bowers (improving their fitness) than less experienced birds. In this work, the best built bower (best position) is intended as an elite iteration. Since the elite position has the highest fitness, it should be able to influence the other positions. The changes of each new bower, representing a new position determined by the position of the best fit bower (position), are calculated according to Eq.(3).

$$x_{ik}^{new} = x_{ik}^{old} + \lambda_k \left(\left(\frac{x_{jk} + x_{elite,k}}{2} \right) - x_{ik}^{old} \right), \tag{3}$$

where x_i is the i^{th} solution vector (bower), x_j is determined as the target solution among all solutions in the current iteration, j is calculated by the roulette wheel procedure, and x_{ik} is the k^{th} member of this dimensions. x_{elite} indicates the elite position (the best fitness value in the current iteration).

$$\lambda_k = \frac{\alpha}{1+p_j} \; . \tag{4}$$

In Eq. (4), λ_k represents the attraction power of the goal bower, shown at intervals of $\lambda_k \in (0, 1)$, where α is the greatest step size (constant) ; and p_j is the probability obtained through Eq.(1) employing the goal bower at intervals of $p_j \in (0, 1)$.

In the mutation process, which occurs at the completion of each iteration of the SBO, random changes are applied with a certain probability. Random changes are then applied to x_{ik} ; again, with a certain probability. The normal distribution (N) within the mutation process is employed through the average of x_{ik}^{old} and the variance of σ^2, as seen in Eqs. (5)-(7).

$$x_{ik}^{new} \sim N(x_{ik}^{old}, \sigma^2), \quad (5)$$

$$N(x_{ik}^{old}, \sigma^2) = x_{ik}^{old} + (\sigma^* N(0,1)), \quad (6)$$

$$\sigma = Z^* (\mathrm{var}_{\max} - \mathrm{var}_{\min}), \quad (7)$$

where σ is a proportion of space width, and $\mathrm{var}_{\min}$ and $\mathrm{var}_{\max}$ are the lower and upper bounds assigned to the variables, respectively. The value of the parameter is the Z percent of the difference between the lower and upper limit, which is variable. In the last of stage of each iteration, the newly formed population and the initial population are evaluated, and all populations are combined and sorted by their fitness values. A new population is then created according to the previously defined number, while the others are rejected.

Chaotic maps for the SBO

Chaos, as a kind of dynamic behavior within a nonlinear chaotic time series, has raised enormous interest in fields such as scientific applications and engineering systems; which have included numerical simulation, chaos control, synchronization, pattern recognition, optimization theory, as well as additional nonlinear sciences. In random-based optimization algorithms, methods employing *chaotic* variables rather than *random* variables are referred to as chaotic optimization algorithms (COA) (Gandomi *et al.*, 2013). Due to the non-repetitiveness of chaotic behavior, the algorithm is capable of carrying out overall searches at higher speeds than stochastic searches, which are dependent upon their probabilities (Hatamlou *et al.*, 2011). One-dimensional, non-invertible maps are the simplest systems capable of generating the desired chaotic motion (Xu *et al.*, 2013).

The proposed CSBO approach

This section presents a novel Chaotic Satin Bowerbird Optimization algorithm called the CSBO, which replaces the main parameter and embeds chaos into the existing SBO. While SBOs possess good convergence rates, they still lack the ability to sufficiently find the global optima, which in turn affects the convergence rate of the algorithm. In order to reduce this effect and to improve its efficiency, the concept of chaos was introduced into the SBO algorithm.

Chaotic maps are imbedded into the SBO to improve the algorithm's solution quality. One of the main parameters of the SBO is the greatest step size (α), which remains a constant parameter. Here, this value (α) is replaced with chaotic maps in an attempt to improve the performance of the SBO. Upon implementation, chaotic maps are normalized between 0 and 1. Furthermore, the parameter of α, determined through Eq. (4), is modified by the chaotic maps through the following equation, Eq. (8).

$$\lambda_k = \begin{cases} \dfrac{c^{t+1}}{1+p_j} & \mathrm{mod}(t, NB-1) = 0, \\ \dfrac{\alpha}{1+p_j} & \text{otherwise} \end{cases} . \quad (8)$$

Where C^{t+1} represents the different chaotic variables, t is the current iteration ($t=1, 2, 3, \ldots m$), m represents the maximum iteration number, and NB is the number of bowers. Eq. (8) produces a design point, from Eq. (3), which uses the different chaotic variables derived from the chaotic maps, with different initial values.

The pseudo-code of the CSBO algorithm is presented in Algorithm 1. In the first step, the bower population within the search space is initialized randomly. After which, the parameters of the CSBO algorithm involved in controlling the exploration and exploitation mechanisms, specifically the NB, P, Z, σ, and α ; are initialized similarly to the SBO. In the second step, the fitness function values of all bowers are initialized in the search space and evaluated using the various standard benchmark functions. The lower fitness value is assumed to be elite (the best fitness function value). The chaotic number of the chaotic map is initialized to adjust parameter α of the SBO. In the third step, the CSBO algorithm runs sequentially, in which all bowers will update their positions, resulting in the first position as the optimal solution. The value of parameter α is also updated along with the course of each iteration through Eq. (8), where $mod(t, NB-1)=0$; t ; is the current iteration, and NB is the population size (Algorithm 1, line 19). In the final step, at the end of the last iteration, the best search agent will be considered as the most optimal solution by the CSBO algorithm.

Algorithm 1 The CSBO algorithm.

Initialize the population size of bowers (NB), greatest step size (α), mutation probability (P), percentage of the difference between the upper and lower limits (Z), proportion of space width (σ) and NFEs=0
Generate the population $X_i=(i=1, 2, 3,..., N)$ of N bowers
For i=1 to N **Do**
Evaluate the fitness value of all bowers $f(X_i)$
The best bower (X_{best}) and assume it as elite
$NFEs=NFEs + NFEs$ that is consumed by bower
End for
// The stage of CSBO
Initial iterations $t=1$
Generate the chaotic sequences $c^t_1 \in (0,1)$, the description in Section 3 $\triangleleft$ (1)
While $(NFEs \geq max\_NFEs)$ **Do**
For $k=1$ to N **Do**
Calculate the probability (P) of bowers using Eqs. (1) and (2)
End for
//Generate a new bower (X_i^{t+1})
For $i=1$ to N **Do**
For $k=1$ to D (all element (D) of bower) **Do**
Select one bower (X_j^t), where (X_j^t) is random using roulette wheel selection
//Calculate step size (λ_k)
If $mod(t, N\text{-}1)=0$ **Then** $\triangleleft$ (2)
Calculate step size (λ_k) using Eq. (8) $\triangleleft$ (3)
Else
Calculate step size (λ_k) using Eq. (4)
End if
Update the position of bower (X_i^{t+1}) using Eq. (3)
//Mutation
If $rand \leq P$ **Then**
Update the position of bower (X_i^{t+1}) using Eq. (6)
End if
End for
Evaluate the fitness value of bower $f(X_i^{t+1})$, $NFEs=NFEs + 1$
End for
Sorted bower (X_N) and $f(X_N)$ by the fitness values
Update elite (X_{best}) if a bower becomes fitter than the elite
$t=t + 1$
End while
Output the global best fitness value of bower (X_{best})

Note: The differences between the SBO and CSBO are indicated with lines marked with the symbol $\triangleleft$ (·).

Experiment results and discussion

In this paper, our experiments were coded in MATLAB R2016a, 64 bit, and run on a desktop computer with an Intel® Core™ i7-6770HQ processor, 8.00GB of RAM, 500GB of HD, and a Microsoft Windows 10 Professional 64 bit Operating System. Moreover, the average objective function values (“Avg.Obj”) and standard deviation of the fitness function values (“Std. Dev”) of all runs were recorded. The “Avg.Obj” and “Std. Dev” were the two performance metrics used to assess the performance of the algorithms.

In this paper, the experiment sets on optimization benchmark problems were implemented to verify the performance of the proposed meta-heuristic CSBO method. Moreover, the CEC2014 (Liang *et al.*, 2014) test suite was selected for the performance evaluation and statistical comparison of the CSBO in the experiments evaluating the performance of the proposed CSBO algorithm in comparison to other meta-heuristic algorithms. Note that all experiments were performed on the same PC, with the same specifications.

1. the performance assessment of CSBO with different chaotic maps

Within the experiment results, the CSBOs utilized the Chebyshev, Circle, Gauss/Mouse, Iterative, Logistic, Piecewise, Sine, Singer, Sinusoidal, and Tent maps (Gandomi *et al*., 2013). The CSBOs were capable of significantly improving the solution quality through the use of chaotic maps. Adjustments of the main parameter α were implemented with the various chaotic maps, as seen in Section 4. It is clear that the number of problems in which better average objective fitness values were obtained by the basic SBO combined with the Tent map, CSBO (Tent map), proving to be superior to all other algorithms. Further detailed testing is reported in Section 5.2.

2. Comparison of the CSBO with other optimization algorithms

To prove the superiority of the proposed CSBO, we conducted comparisons of benchmark problems with 11 well-known algorithms; Crow Search Algorithm (CSA) (Askarzadeh, 2016)we evaluate a novel self-adaptive and auto-constructive metaheuristic called Drone Squadron Optimization (DSO, Firefly Algorithm (FA) (Yang, 2010), Krill Herd algorithm (KH) (Hossein, 2012), Multi-Verse Optimization (MVO) (Mirjalili *et al*., 2016), Whale Optimization Algorithm (WOA) (Mirjalili and Lewis, 2016), Satin Bowerbird Optimizer (SBO), Chaotic Crow Search Algorithm (CCSA), Firefly Algorithm with Chaos (CFA), Chaotic Krill Herd algorithm (CKH), Chaotic Multi-Verse Optimization (CMVO), and Chaotic Whale Optimization Algorithm (CWOA). Thus, in this paper, SBO was considered to be the basic method. The parameter settings for each algorithm in all experiments are shown in sub Section 5.2.1.

2.1 Parameter settings for experiments

We conducted a wide range of tests on the proposed algorithm, benchmarking the performance of the CSBO. Comparisons were made of 11 existing algorithms on 30 benchmark functions, in CEC2014. With a fixed population size of 100 at each run, the benchmark function test problems were executed with 30-D and 50-D, where D is the dimension of the function for all experiments. The experiments were run more than 100 times, in which the maximum number of functions evaluated (*NFEs*) were set at 3.0E+5 and 5.0E+5, respectively. The other main parameters are presented in Table 1.

Table 1 Parameters settings.

#	Chaotic map	Parameters
CSA	-	The awareness probability of crow *AP*=0.1, and the flight length of crow *fl*=2
CCSA	Circle map	
FA	-	The light absorption coefficient=1/L, and the attractiveness coefficient=1
CFA	Sinusoidal map, Gauss/mouse map	The light absorption coefficient=Sinusoidal map, and the attractiveness coefficient=Gauss/mouse map
KH	-	The foraging speed V_f=0.02, the maximum diffusion speed D^{max}=0.005, the maximum induced speed N^{max}=0.01
CKH	Sine map	
MVO	-	The wormhole existence probability WEP_{max}=1, WEP_{min}=0.2, r_1, r_2, r_3 and r_4 are random numbers in [0, 1], and *p*=6
CMVO	Circle map	
WOA	-	*A* and *C* are coefficient vectors, *a* is linearly decreased from 2 to 0, *l* is a random number in [-1, 1], and *p* is a random number in [0, 1]
CWOA	Tent map	
SBO	-	The percentage of the difference between the upper and lower limits () was considered to be 0.02, the mutation probability () was 0.05, and the greatest step size () was 0.94
CSBO	Tent map	

2.2 Numerical results and graphical analysis

In this Section, we present the numerical results, and graphical analysis and evaluation of the CSBO and other optimization algorithms based on the benchmark functions. We further classified the qualitative analyses into four function problems: the Unimodal function, the Simple multimodal function, the Hybrid function, and the Composition function. Tables 2-7 show the statistical analysis of the comparative simulation results between the CSBO and the other algorithms. The four function problems outlined are described as follows:

(a) Unimodal function (f_1-f_3): Table 2 presents the statistical analysis of the simulation results and performs a comparison between the CSBO and the other optimization algorithms. The CSBO algorithm outperformed the comparative algorithms in our statistical tests. Within the 30-dimensional problems, the CSBO generated better simulation results in the functions f_1 and f_2. In cases of the 50-D problems, the CSBO proved superior to all other algorithms in the f_1. However, for functions f_2 and f_3, the CSBO ranked 3rd out of 12 algorithms. Summarizing the overall rank of the 30-D set, the CSBO algorithm ranked first (5.0).

(b) Simple multimodal function (f_4-f_{16}): In the results for the 30-D problems, shown in Table 3, the CSBO algorithm ranked first (f_4, f_{10}-f_{16}) in eight out of 13 benchmark functions. Similar results were achieved in the f_{13} functions for FA, and f_{14} functions for SBO with the CFA algorithms. In the 50-D problem cases, shown in Table 4, the CSBO algorithm was significantly better than or at least similar in f_4, f_{10}-f_{14}, and f_{16} functions in seven out of 13 benchmark functions. The overall rankings (summarized) for the 30-D set found that the CSBO algorithm ranked first (26.0).

(c) Hybrid function (f_{17}-f_{22}): Table 5, in the case of the 30-D problems, the superiority of the CSBO was lost ; and proved inferior to CKH in f_{18}, CMVO and MVO on f_{20}, and FA and CFA in f_{22}. The CSBO algorithm did, however, rank first in three out of the six benchmark functions ; f_{17}, f_{19}, and f_{21}. Within the 50-D set, the CSBO was bettered only by the CFA on f_{22}. In all other areas, the CSBO performed significantly better than the comparative algorithms, ranking first in five out of six benchmark functions (f_{17}-f_{21} benchmark functions). An overall rank summary for the 30-D problems found that the CSBO algorithm ranked first at 8.0.

(d) Composition function (f_{23}-f_{30}): Table 6, within the 30-D problem set, the CSBO ranked less than the CKH and KH on f_{23}-f_{25}, and the CFA, CMVO, MVO, FA, CWOA, and WOA algorithms on f_{28}. However, the CSBO algorithm ranked first in f_{26}, f_{27}, f_{29}, and f_{30} (four out of eight benchmark functions). Within the 50-D problem set, Table 7, the CSBO's performance was inferior to the CKH in f_{23} ; the CWOA, WOA, and CKH on f_{24} ; the CKH, CFA, KH, FA, CMVO, CWOA, and WOA in f_{25} ; the CWOA and WOA on f_{26} ; and to the CFA, CMVO, MVO, FA, CWOA, and WOA in f_{28}. The CSBO ranked first in three out of eight cases ; the f_{27}, f_{29}, and f_{30} benchmark functions). The overall rank summary for the 30-D problems ranked the CSBO algorithm ranks first (23.0), followed by the CFA algorithm (second, at 24.5), and the CKH algorithm (third, at 32.0). Within the 50-D problem set, the CSBO algorithm ranks first (27.5).

The graphical analyses, Figures 1 and 2, display the line graphs of the convergences of all the 12 optimization algorithms, and the 8 functions (f_1, f_4, f_{10}, f_{11}, f_{17}, f_{21}, f_{29}, f_{30}) within the 30-D and 50-D problem set. The CSBO demonstrated better performance in escaping from the local optimum, as well as better search accuracy than the 11 other methods. Moreover, the other algorithms did not find their global optimal value in all of the runs. The values shown in these figures represent the average function optimum achieved from each benchmark function, Tables 2-7.

In summary, the proposed CSBO algorithm achieves better search performance, stable search ability, and a stronger ability to escape from local optimum solutions than all comparative algorithms, presented in Figures 1 and 2 and Tables 2-7. The CSBO method proved to be very efficient for numerical optimization problems. Moreover, the analyses of the non-parametric Wilcoxon's rank sum test, Section 5.2.3 and Section 5.2.4 ; and the Friedman rank test proved the performance of the CSBO algorithm to be superior to all other algorithms tested.

2.3 The non-parametric Wilcoxon's rank sum test

In order to evaluate the performance of proposed CSBO algorithm, we employed the Wilcoxon's rank sum test (Frank Wilcoxon, 1945) to determine the statistical difference of the results achieved by each algorithm. The test was conducted for the results obtained by all algorithms, shown in Tables 2-7, *N/A* indicates "not applicable", denoting the best objective function value in this current function. In the comparison of the CSBO and other optimization algorithms, it is generally considered that a *p*-Value of less than 0.05 indicates that the result achieved by the algorithm is statistically significant, and not obtained by chance. The best results are highlighted in bold face, and the *p*-Values (greater than 0.05) are underlined.

The comparison summaries of the proposed CSBO and other optimization algorithms in the underlined 30-D and 50-D test problems are presented in Tables 2-7. From these tables, it is clear that the number of problems in which better average objective fitness values were obtained by the CSBO algorithm proved it to be superior to all other algorithms. However, the CSBO results are believed to be biased.

2.4 Analysis based on the Friedman rank test

The Friedman rank analyses, present each algorithm was ranked according to their performance using an average Friedman rank competition ranking scheme. In competition ranking, algorithms are put in the same rank if their performances are the same.

Therefore, Figure 3 provides the ranks of 12 optimization algorithms and the overall rank for 30 benchmark functions (Tables 2-7) based on the cases of 30-D (Figure 3a) and 50-D (Figure 3b) mean performances. Using the overall ranks, we can note that the CSBO performs much better than the other algorithms.

Table 2 Unimodal function: comparison of the CSBO with other optimization algorithms for 30-D and 50-D problems. The best Avg.Obj results among the 12 algorithms are shown in bold.

	Method	CSA	FA	KH	MVO	WOA	SBO	CCSA	CFA	CKH	CMVO	CWOA	CSBO
	30-D												
f_1	Avg.Obj	1.6009E+09	2.3063E+07	2.8339E+08	5.0511E+06	3.2328E+07	2.9545E+06	9.5290E+08	4.6221E+06	1.8912E+07	3.4549E+06	2.7226E+07	**3.7548E+05**
	Std. dev.	2.3173E+08	7.5188E+06	9.1114E+07	2.0339E+05	1.4290E+07	8.6304E+05	1.9327E+08	2.4755E+06	5.5471E+06	1.4844E+06	1.1623E+07	**2.0571E+05**
	p-Value	1.7555E-06	7.5569E-10	7.5569E-10	6.6298E-10	7.5569E-10	7.4197E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**
	Rank	12.0	7.0	10.0	5.0	9.0	2.0	11.0	4.0	6.0	3.0	8.0	**1.0**
f_2	Avg.Obj	9.9432E+10	1.1342E+08	1.5577E+10	2.8988E+04	3.0724E+06	5.3763E+05	7.2603E+10	5.5067E+06	2.8004E+04	1.7040E+04	3.0050E+06	**1.1190E+04**
	Std. dev.	7.3771E+09	1.2290E+08	5.6399E+09	1.2094E+04	2.2802E+06	7.8532E+04	1.0275E+10	7.7335E+05	1.7478E+04	1.0766E+04	6.8454E+06	**5.4370E+03**
	p-Value	1.3371E-09	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**
	Rank	12.0	9.0	10.0	4.0	7.0	5.0	11.0	8.0	3.0	2.0	6.0	**1.0**
f_3	Avg.Obj	1.1438E+05	2.4932E+04	8.2161E+04	1.1184E+03	3.7410E+04	1.1468E+04	8.9263E+04	3.4921E+03	6.0446E+04	**4.2596E+02**	3.4955E+04	3.0903E+03
	Std. dev.	1.8373E+04	4.1979E+03	2.4564E+04	3.0058E+02	2.4787E+04	6.2529E+03	2.9399E-11	1.7148E+03	1.3237E+04	**4.0623E+01**	2.2968E+04	2.9297E+03
	p-Value	1.6710E-06	7.5569E-10	1.3025E-09	7.5569E-10	7.5569E-10	7.4669E-10	1.5375E-12	7.5569E-10	7.5569E-10	**N/A**	7.5569E-10	7.5569E-10
	Rank	12.0	6.0	10.0	2.0	8.0	5.0	11.0	4.0	9.0	**1.0**	7.0	3.0
Overall Rank		36.0	22.0	30.0	11.0	24.0	12.0	33.0	16.0	18.0	6.0	21.0	**5.0**
50-D													
f_1	Avg.Obj	7.8524E+09	3.7362E+07	6.2050E+08	7.5993E+06	3.4749E+07	4.4682E+06	3.1273E+09	9.2384E+06	1.2528E+07	7.0843E+06	3.1180E+07	**1.3418E+06**
	Std. dev.	1.8074E+09	1.1120E+07	3.4120E+08	1.7292E+06	1.2664E+07	1.0064E+06	4.0481E+08	3.0593E+06	4.5995E+06	1.9426E+06	1.1635E+07	**3.2025E+05**
	p-Value	2.6254E-09	7.5569E-10	7.5569E-10	7.5228E-10	7.5569E-10	7.4652E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	N/A
	Rank	12.0	9.0	10.0	4.0	8.0	2.0	11.0	5.0	6.0	3.0	7.0	**1.0**

Table 2 Unimodal function: comparison of the CSBO with other optimization algorithms for 30-D and 50-D problems. The best Avg.Obj results among the 12 algorithms are shown in bold. (cont.)

	Method	CSA	FA	KH	MVO	WOA	SBO	CCSA	CFA	CKH	CMVO	CWOA	CSBO
	30-D												
f_2	Avg.Obj	1.9461E+11	3.0454E+09	6.3940E+10	6.8463E+04	2.0063E+07	7.8346E+06	1.7358E+11	4.8625E+06	1.9698E+05	**3.1696E+04**	1.5655E+07	1.9206E+05
	Std. dev.	1.1995E+09	1.2349E+09	1.3058E+10	1.5202E+04	1.4047E+07	7.0536E+05	1.5131E+10	4.7566E+05	1.0358E+05	**1.0989E+04**	1.1330E+07	2.7752E+04
	p-Value	1.4321E-10	7.5569E-10	7.5569E-10	9.2946E-03	7.5569E-10	7.5569E-10	8.5342E-10	7.5569E-10	7.5569E-10	**N/A**	7.5569E-10	5.9268E-06
	Rank	12.0	9.0	10.0	2.0	8.0	6.0	11.0	5.0	4.0	**1.0**	7.0	3.0
f_3	Avg.Obj	3.8902E+05	5.7112E+04	1.5358E+05	4.4193E+02	3.9203E+04	7.7316E+03	2.3116E+05	1.4767E+04	9.6525E+04	**4.2409E+02**	3.4741E+04	1.2720E+03
	Std. dev.	1.6679E+05	6.0507E+03	2.4928E+04	3.7252E+01	8.4108E+03	3.5901E+03	3.0561E+04	4.8265E+03	1.7437E+04	**3.0203E+01**	7.8181E+03	6.1448E+02
	p-Value	4.4668E-11	7.5569E-10	7.5569E-10	2.9456E-04	7.5569E-10	7.3711E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**	7.5569E-10	7.5569E-10
	Rank	12.0	8.0	10.0	2.0	7.0	4.0	11.0	5.0	9.0	**1.0**	6.0	3.0
Overall rank		36.0	26.0	30.0	8.0	23.0	12.0	33.0	15.0	19.0	**5.0**	20.0	7.0

Table 3 Simple Multimodal functions: comparison of the CSBO with other optimization algorithms for 30-D problems. The best Avg.Obj results among the 12 algorithms are shown in bold.

	Method	CSA	FA	KH	MVO	WOA	SBO	CCSA	CFA	CKH	CMVO	CWOA	CSBO
	30-D												
f_4	Avg.Obj	2.3316E+04	5.9775E+02	1.6912E+03	5.0136E+02	5.8132E+02	5.3658E+02	1.2597E+04	5.1262E+02	5.0141E+02	4.9630E+02	5.7865E+02	**4.5293E+02**
	Std. dev.	2.5274E+03	3.8802E+01	7.3894E+02	2.5362E+01	5.8743E+01	2.4539E+01	2.3726E+03	2.2069E+01	3.1247E+01	3.3965E+01	4.6182E+01	**3.5539E+01**
	p-Value	1.1101E-09	7.5569E-10	7.5569E-10	7.5399E-10	7.5569E-10	7.4230E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**
	Rank	12.0	9.0	10.0	3.0	8.0	6.0	11.0	5.0	4.0	2.0	7.0	**1.0**
f_5	Avg.Obj	5.2098E+02	5.2094E+02	**5.2001E+02**	5.2007E+02	5.2032E+02	5.2085E+02	5.2095E+02	5.2094E+02	**5.2001E+02**	5.2005E+02	5.2020E+02	5.2004E+02
	Std. dev.	6.3737E-02	4.8568E-02	**2.9419E-03**	5.3683E-02	1.5986E-01	7.4616E-02	5.6665E-02	4.8739E-02	**5.1790E-03**	4.0495E-02	1.8728E-01	1.7613E-02
	p-Value	7.0957E-07	7.6048E-03	**N/A**	7.5569E-10	7.5569E-10	5.8601E-09	1.2009E-03	2.8109E-03	**N/A**	7.5569E-10	7.5569E-10	5.5448E-03
	Rank	12.0	9.5	**1.5**	5.0	7.0	8.0	11.0	9.5	**1.5**	4.0	6.0	3.0
f_6	Avg.Obj	6.4571E+02	6.1140E+02	6.3583E+02	6.1048E+02	6.3574E+02	6.2847E+02	6.4075E+02	**6.0684E+02**	6.2093E+02	6.1011E+02	6.3521E+02	6.1978E+02
	Std. dev.	1.4259E+00	1.9603E+00	2.2431E+00	2.0173E+00	2.7854E+00	1.2209E+00	1.1875E+00	**2.3952E+00**	3.1404E+00	2.9034E+00	3.3222E+00	1.0952E+00
	p-Value	1.1812E-09	7.5569E-10	7.5569E-10	7.5228E-10	7.5569E-10	7.0596E-10	7.5569E-10	**N/A**	7.5569E-10	7.5569E-10	7.5569E-10	7.0516E-10
	Rank	12.0	4.0	10.0	3.0	9.0	7.0	11.0	**1.0**	6.0	2.0	8.0	5.0
f_7	Avg.Obj	1.5135E+03	7.0294E+02	8.7406E+02	7.0009E+02	7.0101E+02	7.0093E+02	1.3546E+03	7.0105E+02	**7.0002E+02**	7.0008E+02	7.0090E+02	7.0007E+02
	Std. dev.	5.0219E+01	1.6586E+00	6.6743E+01	3.3399E-02	6.8074E-02	3.5790E-02	6.6157E+01	6.5845E-03	**8.9813E-03**	1.0499E-02	1.1995E-01	2.0174E-02
	p-Value	3.2718E-01	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**	7.5569E-10	7.5569E-10	3.3041E-01
	Rank	12.0	9.0	10.0	4.0	7.0	6.0	11.0	8.0	**1.0**	3.0	5.0	2.0
f_8	Avg.Obj	1.1741E+03	8.5792E+02	9.5934E+02	8.8281E+02	9.8531E+02	9.4291E+02	1.1696E+03	**8.4189E+02**	9.1184E+02	8.8182E+02	9.8134E+02	8.8059E+02
	Std. dev.	8.8788E+00	8.4674E+00	2.2284E+01	2.0580E+01	3.9847E+01	2.6164E+01	1.5480E+01	**1.0384E+01**	2.3041E+01	2.0337E+01	3.7717E+01	5.0253E+00
	p-Value	6.0845E-08	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	8.8418E-03	**N/A**	7.5569E-10	7.5569E-10	7.5569E-10	2.1636E-03
	Rank	12.0	2.0	8.0	5.0	10.0	7.0	11.0	**1.0**	6.0	4.0	9.0	3.0
f_9	Avg.Obj	1.3572E+03	9.6453E+02	1.0919E+03	1.0226E+03	1.1364E+03	1.0877E+03	1.3393E+03	**9.5379E+02**	1.0218E+03	1.0206E+03	1.1238E+03	1.0177E+03
	Std. dev.	9.3912E+00	1.3016E+01	2.3571E+01	3.4674E+00	6.0934E+01	2.8292E+01	2.2771E+01	**1.6183E+01**	2.0375E+01	2.9034E+01	5.6711E+01	3.1464E-03
	p-Value	2.2503E-07	7.5569E-02	7.5569E-10	5.1059E-10	7.5569E-10	7.5569E-10	8.3138E-02	**N/A**	7.5569E-10	7.5569E-10	7.5569E-10	2.1636E-03
	Rank	12.0	2.0	8.0	6.0	10.0	7.0	11.0	**1.0**	5.0	4.0	9.0	3.0

Table 3 Simple Multimodal functions: comparison of the CSBO with other optimization algorithms for 30-D problems. The best Avg.Obj results among the 12 algorithms are shown in bold. (cont.)

Method		CSA	FA	KH	MVO	WOA	SBO	CCSA	CFA	CKH	CMVO	CWOA	CSBO
30-D													
f_{10}	Avg.Obj	8.6963E+03	3.2277E+03	4.7968E+03	3.9748E+03	5.0048E+03	3.8015E+03	8.1082E+03	3.1170E+03	4.4174E+03	3.9213E+03	4.9901E+03	**3.0895E+03**
	Std. dev.	3.5537E+02	5.2523E+02	6.5854E+02	6.9957E+02	6.3017E+02	4.9797E+02	3.4483E+02	5.0187E+02	7.8190E+02	7.3778E+02	6.1531E+02	**2.4836E+02**
	p-Value	2.1093E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	3.7850E-09	7.5552E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**
	Rank	12.0	3.0	8.0	6.0	10.0	4.0	11.0	2.0	7.0	5.0	9.0	**1.0**
f_{11}	Avg.Obj	9.4018E+03	4.0725E+03	5.4627E+03	4.3587E+03	5.9709E+03	5.0707E+03	8.5721E+03	4.0597E+03	5.0243E+03	4293.74917	5.8132E+03	**3.9633E+03**
	Std. dev.	3.4139E+02	4.0912E+02	6.7747E+02	6.5400E+02	9.5793E+02	5.8713E+02	2.5112E+02	4.8890E+02	5.3995E+02	6.7703E+02	7.3240E+02	**1.0387E+02**
	p-Value	4.2323E-10	7.5569E-10	7.5569E-10	7.5484E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**
	Rank	12.0	3.0	8.0	5.0	10.0	7.0	11.0	2.0	6.0	4.0	9.0	**1.0**
f_{12}	Avg.Obj	1.2034E+03	1.2025E+03	1.2007E+03	1.2003E+03	1.2016E+03	1.2002E+03	1.2027E+03	1.2003E+03	1.2003E+03	1.2003E+03	1.2016E+03	**1.2001E+03**
	Std. dev.	4.1543E-01	2.5325E-01	2.7726E-01	2.1034E-01	4.1465E-01	2.1980E-02	3.0415E-01	3.7214E-02	1.6224E-01	1.3379E-01	4.5029E-01	**5.8079E-02**
	p-Value	5.0452E-09	7.5569E-10	7.5569E-10	7.5484E-10	7.5569E-10	7.5484E-05	2.1002E-09	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**
	Rank	12.0	10.0	7.0	4.5	8.5	2.0	11.0	4.5	4.5	4.5	8.5	**1.0**
f_{13}	Avg.Obj	1.3100E+03	**1.3003E+03**	1.3036E+03	1.3004E+03	1.3005E+03	1.3005E+03	1.3071E+03	**1.3003E+03**	1.3004E+03	1.3004E+03	1.3005E+03	**1.3003E+03**
	Std. dev.	9.2056E-01	**4.7152E-02**	7.8677E-01	1.0193E-01	1.1919E-01	1.0454E-01	4.8467E-01	**3.8477E-02**	7.9592E-02	9.8132E-02	1.3845E-01	**5.2780E-02**
	p-Value	7.7397E-08	**N/A**	7.5569E-10	7.5569E-10	7.5569E-10	7.4702E-10	7.5569E-10	**N/A**	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**
	Rank	12.0	**2.0**	10.0	5.0	8.0	8.0	11.0	**2.0**	5.0	5.0	8.0	**2.0**
f_{14}	Avg.Obj	1.6730E+03	1.4003E+03	1.4698E+03	1.4005E+03	1.4003E+03	**1.4002E+03**	1.6288E+03	**1.4002E+03**	1.4003E+03	1.4004E+03	1.4003E+03	**1.4002E+03**
	Std. dev.	1.1890E+01	4.1027E-02	1.8664E+01	3.0269E-01	5.9385E-02	**4.8548E-02**	2.5400E+01	**2.7460E-02**	7.7811E-02	2.9025E-01	1.0815E-01	**2.5058E-02**
	p-Value	5.2113E-09	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**	9.0681E-10	**N/A**	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**
	Rank	12.0	5.5	10.0	9.0	5.5	**2.0**	11.0	**2.0**	5.5	8.0	5.5	**2.0**
f_{15}	Avg.Obj	1.2050E+06	1.5146E+03	1.5398E+03	1.5105E+03	1.5753E+03	1.5312E+03	2.7359E+05	1.5136E+03	1.5199E+03	1.5101E+03	1.5747E+03	**1.5096E+03**
	Std. dev.	5.0473E+05	1.4775E+00	7.9678E+00	1.3555E+00	2.7361E+01	5.8988E+00	8.5609E+03	2.5566E+00	5.0161E+00	1.1168E+00	2.6867E+01	**8.8795E-01**
	p-Value	1.3129E-11	7.5569E-10	7.5569E-10	7.3878E-10	7.5569E-10	6.8129E-10	2.8249E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**
	Rank	12.0	5.0	8.0	3.0	10.0	7.0	11.0	4.0	6.0	2.0	9.0	**1.0**
f_{16}	Avg.Obj	1.6125E+03	1.6115E+03	1.6128E+03	1.6117E+03	1.6127E+03	1.6122E+03	1.6121E+03	1.6114E+03	1.6125E+03	1.6116E+03	1.6125E+03	**1.6113E+03**
	Std. dev.	3.3965E-01	4.2268E-01	3.9383E-01	3.4767E-01	4.3054E-01	5.8897E-01	3.0829E-01	4.5936E-13	4.2138E-01	7.1779E-01	5.4337E-01	**7.0681E-01**
	p-Value	5.7407E-09	7.5569E-10	4.3371E-10	7.5399E-10	1.8355E-04	1.2094E-09	8.0311E-10	1.5375E-12	1.9919E-06	8.0311E-10	3.3508E-06	**N/A**
	Rank	9.0	3.0	12.0	5.0	11.0	7.0	6.0	2.0	9.0	4.0	9.0	**1.0**
Overall rank		153.0	67.0	110.5	63.5	114.0	78.0	138.0	44.0	66.5	51.5	102.0	**26.0**

Table 4 Simple Multimodal functions: comparison of the CSBO with other optimization algorithms for 50-D problems. The best Avg.Obj results among the 12 algorithms are shown in bold.

	Method	CSA	FA	KH	MVO	WOA	SBO	CCSA	CFA	CKH	CMVO	CWOA	CSBO
	50-D												
f_4	Avg.Obj	5.8462E+04	1.0308E+03	1.0766E+04	5.3067E+02	6.5999E+02	5.9715E+02	4.7366E+04	5.0637E+02	5.4695E+02	5.0755E+02	6.4454E+02	**4.9020E+02**
	Std. dev.	2.1042E+03	1.3299E+02	3.3915E+03	5.1473E+00	6.5643E+01	4.4552E+01	6.7688E+03	1.7025E+01	4.4424E+01	2.8943E+01	6.3439E+01	**4.4169E+01**
	p-Value	4.2507E-11	7.5569E-10	7.5569E-10	2.1636E-10	7.5569E-10	7.4584E-10	2.5085E-09	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**
	Rank	12.0	9.0	10.0	4.0	8.0	6.0	11.0	2.0	5.0	3.0	7.0	**1.0**
f_5	Avg.Obj	5.2117E+02	5.2113E+02	5.2002E+02	5.2014E+02	5.2046E+02	5.2105E+02	5.2115E+02	5.2106E+02	**5.2000E+02**	5.2013E+02	5.2032E+02	5.2011E+02
	Std. dev.	3.5276E-02	3.8638E-02	3.3088E-03	7.8409E-02	1.5677E-01	3.1402E-02	3.7536E-02	4.1462E-02	**9.7675E-04**	5.1228E-02	2.1935E-01	2.2968E-13
	p-Value	3.3136E-11	7.5569E-10	5.5569E-03	4.3816E-05	7.5569E-10	7.4753E-10	2.5085E-09	7.5569E-10	**N/A**	7.5569E-10	7.5569E-10	1.5375E-05
	Rank	12.0	10.0	2.0	5.0	7.0	8.0	11.0	9.0	**1.0**	4.0	6.0	3.0
f_6	Avg.Obj	6.7940E+02	6.2784E+02	6.6640E+02	6.3434E+02	6.6549E+02	6.5601E+02	6.7380E+02	6.2730E+02	6.4440E+02	**6.2511E+02**	6.6442E+02	6.2974E+02
	Std. dev.	1.9365E+00	2.8436E+00	3.0942E+00	2.0238E+00	3.7860E+00	2.8107E+00	1.4055E+00	2.6007E+00	4.3959E+00	**4.6030E+00**	5.3037E+00	4.5936E-13
	p-Value	1.8756E-11	7.5569E-05	7.5569E-10	6.2597E-10	7.5569E-10	7.5399E-10	7.5569E-10	2.9456E-05	7.5569E-10	**N/A**	7.5569E-10	1.5375E-05
	Rank	12.0	3.0	10.0	5.0	9.0	7.0	11.0	2.0	6.0	**1.0**	8.0	4.0
f_7	Avg.Obj	2.4704E+03	7.3667E+02	1.3383E+03	7.0019E+02	7.0121E+02	7.0112E+02	2.3664E+03	7.0104E+02	**7.0006E+02**	7.0011E+02	7.0118E+02	7.0043E+02
	Std. dev.	3.4407E+01	9.9674E+00	1.4886E+02	4.2986E-02	1.1989E-01	9.7072E-03	1.1796E+02	6.0621E-03	**3.0629E-02**	2.3081E-02	1.1286E-01	4.1054E-02
	p-Value	4.0100E-05	7.5569E-10	7.5569E-10	8.6929E-03	7.5569E-10	7.5569E-10	3.0382E-07	7.5569E-10	**N/A**	6.9295E-03	7.5569E-10	8.2207E-03
	Rank	12.0	9.0	10.0	3.0	8.0	6.0	11.0	5.0	**1.0**	2.0	7.0	4.0
f_8	Avg.Obj	1.5370E+03	9.5000E+02	1.1606E+03	9.6275E+02	1.1407E+03	1.0999E+03	1.4716E+03	**8.9409E+02**	1.0454E+03	9.5937E+02	1.1289E+03	1.0343E+03
	Std. dev.	2.4535E+01	1.8385E+01	3.7446E+01	3.4164E+01	6.4791E+01	3.4908E+01	1.0134E+01	**1.8626E+01**	2.9780E+01	3.9404E+01	5.9020E+01	6.0118E+00
	p-Value	4.0937E-11	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**	7.5569E-10	7.5569E-10	7.5569E-10	6.4424E-10
	Rank	12.0	2.0	10.0	4.0	9.0	7.0	11.0	**1.0**	6.0	3.0	8.0	5.0
f_9	Avg.Obj	1.7897E+03	1.0550E+03	1.3628E+03	1.2090E+03	1.3343E+03	1.3185E+03	1.7787E+03	**1.0514E+03**	1.1664E+03	1.0970E+03	1.3103E+03	1.1893E+03
	Std. dev.	6.5833E+00	2.0946E+01	3.9892E+01	2.0932E+01	7.6689E+01	3.5148E+01	4.6222E+01	**2.3337E+01**	4.0233E+01	3.9711E+01	6.6587E+01	3.4428E+00
	p-Value	7.5569E-10	7.5569E-10	7.5569E-10	3.4580E-10	7.5569E-10	7.5262E-10	1.2363E-01	**N/A**	7.5569E-10	7.5569E-10	7.5569E-10	6.2597E-10
	Rank	12.0	2.0	10.0	6.0	9.0	8.0	11.0	**1.0**	4.0	3.0	7.0	5.0
f_{10}	Avg.Obj	1.5544E+04	6.3038E+03	8.9981E+03	7.0248E+03	8.5387E+03	6.2091E+03	1.4567E+04	6.2982E+03	8.0166E+03	6.6706E+03	8.4090E+03	**5.2472E+03**
	Std. dev.	4.5485E+02	7.6119E+02	1.0347E+03	9.9581E+02	1.1006E+03	7.3454E+02	3.3691E+02	6.8874E+02	1.1391E+03	9.1759E+02	1.2410E+03	**5.0984E+02**
	p-Value	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**
	Rank	12.0	4.0	10.0	6.0	9.0	2.0	11.0	3.0	7.0	5.0	8.0	**1.0**
f_{11}	Avg.Obj	1.6090E+04	6.8120E+03	9.4698E+03	7.0628E+03	1.0052E+04	8.2681E+03	1.4992E+04	6.1131E+03	7.8356E+03	6.8517E+03	9.5257E+03	**6.0896E+03**
	Std. dev.	4.1120E+02	7.7287E+02	1.1157E+03	7.8431E+02	1.2692E+03	7.8569E+02	2.8252E+02	8.1745E+02	1.0576E+03	8.4969E+02	1.3690E+03	**9.4643E+01**
	p-Value	7.6159E-11	7.5569E-10	7.5569E-10	7.5228E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**
	Rank	12.0	3.0	8.0	5.0	10.0	7.0	11.0	2.0	6.0	4.0	9.0	**1.0**

Table 4 Simple Multimodal functions: comparison of the CSBO with other optimization algorithms for 50-D problems. The best Avg.Obj results among the 12 algorithms are shown in bold. (cont.)

Method		CSA	FA	KH	MVO	WOA	SBO	CCSA	CFA	CKH	CMVO	CWOA	CSBO
50-D													
f_{12}	Avg.Obj	1.2043E+03	1.2033E+03	1.2009E+03	1.2004E+03	1.2024E+03	1.2005E+03	1.2036E+03	**1.2001E+03**	1.2004E+03	1.2004E+03	1.2023E+03	**1.2001E+03**
	Std. dev.	4.2839E-01	2.9498E-01	3.6092E-01	2.2307E-01	6.0490E-01	5.2558E-02	3.4745E-01	**1.9480E-02**	2.4395E-01	1.8684E-01	5.8115E-01	**4.2965E-02**
	p-Value	1.6524E-11	7.5569E-10	7.5569E-10	7.5484E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**
	Rank	12.0	10.0	7.0	4.0	9.0	6.0	11.0	**1.5**	4.0	4.0	8.0	**1.5**
f_{13}	Avg.Obj	1.3097E+03	1.3005E+03	1.3053E+03	1.3006E+03	1.3006E+03	1.3006E+03	1.3091E+03	1.3005E+03	1.3005E+03	1.3006E+03	1.3005E+03	**1.3004E+03**
	Std. dev.	1.2459E-01	3.4525E-02	5.7142E-01	1.0510E-01	1.0673E-01	6.1996E-02	4.1906E-01	4.4276E-02	7.5928E-02	1.2628E-01	1.1331E-01	**5.3808E-02**
	p-Value	1.4701E-11	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5399E-10	1.2265E-09	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**
	Rank	12.0	3.5	10.0	7.5	7.5	7.5	11.0	3.5	3.5	7.5	3.5	**1.0**
f_{14}	Avg.Obj	1.8370E+03	1.4004E+03	1.5483E+03	1.4006E+03	1.4004E+03	1.4003E+03	1.7646E+03	1.4003E+03	1.4003E+03	1.4006E+03	1.4003E+03	**1.4002E+03**
	Std. dev.	3.7086E+01	2.0015E-01	2.9510E+01	3.8182E-01	8.8102E-02	3.2112E-02	1.6158E+01	2.0301E-02	3.5752E-02	3.9897E-01	4.3883E-02	**2.5043E-02**
	p-Value	5.2761E-11	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5467E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**
	Rank	12.0	6.5	10.0	8.5	6.5	3.5	11.0	3.5	3.5	8.5	3.5	**1.0**
f_{15}	Avg.Obj	1.6512E+07	1.5926E+03	4.2001E+03	1.5244E+03	1.7616E+03	1.5462E+03	1.1949E+07	1.5221E+03	1.5576E+03	**1.5163E+03**	1.7588E+03	1.5203E+03
	Std. dev.	1.9193E+06	2.7541E+01	1.0592E+04	3.9252E+00	7.2849E+01	6.2357E+00	4.0923E+06	2.6775E+00	1.2470E+01	**3.9272E+00**	6.8699E+01	1.0743E+00
	p-Value	3.6528E-11	7.5569E-10	7.5569E-10	5.4142E-10	7.5569E-10	7.4770E-10	2.0979E-11	6.2946E-05	7.5569E-10	**N/A**	7.5569E-10	2.1636E-02
	Rank	12.0	7.0	10.0	4.0	9.0	5.0	11.0	3.0	6.0	**1.0**	8.0	2.0
f_{16}	Avg.Obj	1.6223E+03	1.6209E+03	1.6220E+03	1.6210E+03	1.6224E+03	1.6216E+03	1.6218E+03	1.6200E+03	1.6220E+03	1.6210E+03	1.6222E+03	**1.6199E+03**
	Std. dev.	3.3958E-01	7.0150E-01	4.1961E-01	6.4453E-01	5.1830E-01	7.1383E-01	3.3830E-01	2.9187E-01	3.1933E-01	6.2692E-01	4.5614E-01	**5.8796E-01**
	p-Value	1.1776E-09	9.0681E-03	2.2571E-09	1.1536E-09	2.9108E-06	1.5164E-08	7.8262E-08	5.5293E-03	9.3460E-10	1.5591E-09	5.1977E-09	**N/A**
	Rank	11.0	3.0	8.5	4.5	12.0	6.0	7.0	2.0	8.5	4.5	10.0	**1.0**
Overall rank		155.0	72.0	115.5	66.5	113.0	79.0	139.0	38.5	61.5	50.5	93.0	30.5

Table 5 Hybrid functions: comparison of the CSBO with other optimization algorithms for 30-D and 50-D problems. The best Avg.Obj results among the 12 algorithms are shown in bold.

Method		CSA	FA	KH	MVO	WOA	SBO	CCSA	CFA	CKH	CMVO	CWOA	CSBO
30-D													
f_{17}	Avg.Obj	2.3799E+08	3.8818E+05	1.0118E+07	2.5099E+05	3.5065E+06	8.2220E+05	3.6996E+07	3.4212E+05	1.7306E+06	2.2422E+05	2.6037E+06	**9.8969E+04**
	Std. dev.	9.4108E+07	2.6890E+05	6.7400E+06	1.4745E+05	2.3569E+06	2.7776E+05	1.0321E+07	2.0950E+05	1.2565E+06	1.4323E+05	1.4260E+06	**7.3983E+04**
	p-Value	3.2868E-09	7.5569E-10	7.5569E-10	7.5228E-10	7.5569E-10	7.4669E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**
	Rank	12.0	5.0	10.0	3.0	9.0	6.0	11.0	4.0	7.0	2.0	8.0	**1.0**
f_{18}	Avg.Obj	7.7705E+09	7.9165E+04	2.2474E+07	1.7227E+04	1.2282E+04	4.7063E+03	1.7846E+09	6.5200E+04	**3.0415E+03**	1.4159E+04	1.0142E+04	3.9436E+03
	Std. dev.	2.4021E+09	2.3140E+04	9.4741E+07	9.7142E+03	32335.5	3.3311E+03	5.6451E+08	1.5120E+04	**1.5712E+03**	1.0984E+04	1.8094E+04	2.6935E+03
	p-Value	5.8098E-11	7.5569E-10	7.5569E-10	7.5399E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**	7.5569E-10	7.5569E-10	2.1579E-03
	Rank	12.0	9.0	10.0	7.0	5.0	3.0	11.0	8.0	**1.0**	6.0	4.0	2.0

Table 5 Hybrid functions: comparison of the CSBO with other optimization algorithms for 30-D and 50-D problems. The best Avg.Obj results among the 12 algorithms are shown in bold. (cont.)

Method		CSA	FA	KH	MVO	WOA	SBO	CCSA	CFA	CKH	CMVO	CWOA	CSBO
30-D													
f_{19}	Avg.Obj	2.7328E+03	1.9162E+03	1.9275E+03	1.9144E+03	1.9442E+03	1.9658E+03	2.2677E+03	1.9159E+03	1.9158E+03	1.9127E+03	1.9431E+03	**1.9114E+03**
	Std. dev.	1.0448E+02	7.8577E+00	2.5256E+01	1.3954E+01	37.8	1.6300E+01	5.2296E+01	8.6308E+00	8.5975E+00	1.0784E+01	3.2127E+01	**1.8808E+00**
	p-Value	9.6052E-11	7.5569E-10	7.5569E-10	7.5484E-10	7.5569E-10	6.8144E-10	7.5569E-10	7.5552E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**
	Rank	12.0	6.0	7.0	3.0	9.0	10.0	11.0	5.0	4.0	2.0	8.0	**1.0**
f_{20}	Avg.Obj	2.6840E+06	1.0761E+04	6.3547E+04	2.3987E+03	2.5788E+04	1.6249E+04	1.7309E+05	3.4584E+03	2.4095E+04	**2.3780E+03**	2.0685E+04	3.0813E+03
	Std. dev.	1.9727E+06	3.3144E+03	4.4469E+04	1.1336E+02	1.5127E+04	7.8963E+03	9.0965E+04	1.0807E+03	1.0348E+04	**1.1426E+02**	1.2932E+04	4.3954E+02
	p-Value	3.6484E-06	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5160E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**	7.5569E-10	7.2220E-08
	Rank	12.0	5.0	10.0	2.0	9.0	6.0	11.0	4.0	8.0	**1.0**	7.0	3.0
f_{21}	Avg.Obj	8.8824E+07	9.2658E+04	5.4296E+06	7.6027E+04	1.1376E+06	5.4364E+05	1.1684E+07	8.9538E+04	4.8133E+05	7.3129E+04	9.5673E+05	**7.2886E+04**
	Std. dev.	5.0304E+07	8.1384E+04	4.4055E+06	3.3994E+04	9.5708E+05	2.6735E+05	4.7384E+06	5.1668E+04	4.0490E+05	3.4622E+04	6.7945E+05	**4.5934E+04**
	p-Value	5.7773E-11	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5024E-10	7.5569E-10	7.5245E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**
	Rank	12.0	5.0	10.0	3.0	9.0	7.0	11.0	4.0	6.0	2.0	8.0	**1.0**
f_{22}	Avg.Obj	1.4461E+04	2.4501E+03	3.1466E+03	2.5714E+03	3.0278E+03	3.4264E+03	3.9432E+03	**2.4466E+03**	2.8155E+03	2.5355E+03	3.0025E+03	2.4858E+03
	Std. dev.	1.7645E+04	6.7729E+01	2.3560E+02	1.5870E+02	2.0829E+02	2.0830E+02	2.1747E+02	**7.7481E+01**	2.0854E+02	1.5982E+02	2.1972E+02	1.9248E+01
	p-Value	3.2813E-11	7.5569E-10	7.5569E-10	7.5484E-10	7.5569E-10	7.3811E-10	7.5569E-10	**N/A**	7.5569E-10	7.5569E-10	7.5569E-10	2.1636E-10
	Rank	12.0	2.0	9.0	5.0	8.0	10.0	11.0	**1.0**	6.0	4.0	7.0	3.0
f_{17}	Avg.Obj	9.7588E+08	8.6491E+05	4.9362E+07	6.7084E+05	1.8098E+07	1.2048E+06	2.3809E+08	8.4569E+05	2.3871E+06	5.8095E+05	1.4128E+07	**1.3032E+05**
	Std. dev.	3.0751E+08	4.3737E+05	3.5157E+07	2.4522E+05	1.0542E+07	3.3170E+05	6.1997E+07	5.0180E+05	1.2062E+06	2.3867E+05	8.5281E+06	**4.7015E+04**
	p-Value	7.5569E-10	7.5569E-10	7.5569E-10	7.5228E-10	7.5569E-10	7.5399E-10	7.5569E-10	7.5569E-10	7.5569E-10	2.3585E-06	7.5569E-10	**N/A**
	Rank	12.0	5.0	10.0	3.0	9.0	6.0	11.0	4.0	7.0	2.0	8.0	**1.0**
f_{18}	Avg.Obj	2.6544E+10	8.4821E+04	1.3580E+09	5.9004E+03	1.0708E+04	5.7690E+03	1.0299E+10	7.4520E+04	3.5677E+03	5.2529E+03	5.3358E+03	**3.2621E+03**
	Std. dev.	4.1259E+09	1.3916E+04	1.4553E+09	2.1625E+03	8.0898E+03	1.2376E+03	1.6891E+09	1.0210E+04	1.2783E+03	2.1504E+03	2.0796E+03	**1.4043E+03**
	p-Value	7.5569E-10	7.5569E-10	7.5569E-10	7.5228E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	4.8591E-06	2.5693E-08	7.5569E-10	**N/A**
	Rank	12.0	9.0	10.0	6.0	7.0	5.0	11.0	8.0	2.0	3.0	4.0	**1.0**
f_{19}	Avg.Obj	6.7867E+03	1.9555E+03	1.9828E+03	1.9242E+03	1.9863E+03	1.9681E+03	3.2330E+03	1.9528E+03	1.9479E+03	1.9239E+03	1.9822E+03	**1.9232E+03**
	Std. dev.	1.3737E+03	2.7468E+01	1.2040E+02	2.7284E+00	3.0161E+01	3.1111E+01	2.0942E+02	2.6465E+01	2.9197E+01	9.7989E+00	3.0167E+01	**8.5602E+00**
	p-Value	7.5569E-10	7.5569E-10	7.5569E-10	7.4382E-10	7.5569E-10	7.5416E-10	7.5569E-10	7.5211E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**
	Rank	12.0	6.0	9.0	3.0	10.0	7.0	11.0	5.0	4.0	2.0	8.0	**1.0**
f_{20}	Avg.Obj	1.4865E+06	1.1760E+04	1.1124E+05	2.7026E+03	9.5643E+04	9.8378E+03	3.9347E+05	4.1285E+03	2.8924E+04	2.6891E+03	8.2767E+04	**2.6531E+03**
	Std. dev.	7.1230E+02	2.8654E+03	6.5140E+04	1.1786E+02	9.0336E+04	4.7445E+03	1.5942E+05	8.8219E+02	9.5299E+03	1.3966E+02	6.1238E+04	**5.0937E+02**
	p-Value	7.5569E-10	7.5569E-10	7.5569E-10	7.5552E-10	7.5569E-10	7.5228E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**
	Rank	12.0	6.0	10.0	3.0	9.0	5.0	11.0	4.0	7.0	2.0	8.0	**1.0**
f_{21}	Avg.Obj	3.5227E+08	4.9253E+05	1.1504E+07	4.4711E+05	5.1710E+06	8.3068E+05	5.5107E+07	4.2194E+05	2.6425E+06	4.0437E+05	4.5187E+06	**9.5857E+04**
	Std. dev.	1.4564E+08	3.1941E+05	5.9151E+06	1.5025E+05	3.1599E+06	3.2473E+05	1.3778E+07	2.0779E+05	1.1071E+06	1.6968E+05	3.3718E+06	**4.1897E+04**
	p-Value	7.5569E-10	7.5569E-10	7.5569E-10	7.5279E-10	7.5569E-10	7.5484E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**
	Rank	12.0	5.0	10.0	4.0	9.0	6.0	11.0	3.0	7.0	2.0	8.0	**1.0**

Table 5 Hybrid functions: comparison of the CSBO with other optimization algorithms for 30-D and 50-D problems. The best Avg.Obj results among the 12 algorithms are shown in bold. (cont.)

Method		CSA	FA	KH	MVO	WOA	SBO	CCSA	CFA	CKH	CMVO	CWOA	CSBO
30-D													
f_{22}	Avg.Obj	4.9372E+05	2.8717E+03	4.2585E+03	3.1872E+03	4.2343E+03	4.2081E+03	8.2666E+03	**2.8188E+03**	3.6787E+03	3.1199E+03	4.0302E+03	3.0674E+03
	Std. dev.	4.8462E+05	3.1929E+02	5.3113E+02	2.5854E+02	4.9537E+02	3.2813E+02	1.6985E+03	**1.8963E+02**	3.9958E+02	3.1231E+02	4.1731E+02	2.5273E+02
	p-Value	7.5569E-10	4.2946E-03	7.5569E-10	7.3878E-10	7.5569E-10	7.5399E-10	7.5569E-10	**N/A**	7.5569E-10	7.5569E-10	7.5569E-10	1.5375E-03
	Rank	12.0	2.0	10.0	5.0	9.0	8.0	11.0	**1.0**	6.0	4.0	7.0	3.0
Overall rank		72.0	33.0	59.0	24.0	53.0	37.0	66.0	25.0	33.0	15.0	43.0	**8.0**

Table 6 Composition functions: comparison of the CSBO with other optimization algorithms for 30-D problems. The best Avg.Obj results among the 12 algorithms are shown in bold.

Method		CSA	FA	KH	MVO	WOA	SBO	CCSA	CFA	CKH	CMVO	CWOA	CSBO
30-D													
f_{23}	Avg.Obj	3.1380E+03	2.6222E+03	2.6151E+03	2.6158E+03	2.6354E+03	2.6153E+03	3.1264E+03	2.6153E+03	**2.6114E+03**	2.6157E+03	2.6337E+03	2.6152E+03
	Std. dev.	1.1085E+02	2.1393E+00	8.2920E+01	4.6064E-01	1.1729E+01	1.7565E-03	1.1083E+02	1.2428E-02	**2.2992E+01**	4.7429E-01	9.9095E+00	4.4453E-04
	p-Value	2.1108E-11	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	1.6227E-03	7.5569E-10	**N/A**	7.5569E-10	7.5569E-10	7.5569E-10
	Rank	12.0	8.0	2.0	7.0	10.0	4.5	11.0	4.5	**1.0**	6.0	9.0	3.0
f_{24}	Avg.Obj	2.7877E+03	2.6256E+03	2.6105E+03	2.6315E+03	2.6073E+03	2.6509E+03	2.7872E+03	2.6248E+03	**2.6018E+03**	2.6310E+03	2.6067E+03	2.6051E+03
	Std. dev.	1.6495E+01	1.2132E+00	1.0718E+01	7.8199E+00	4.5192E+00	1.7631E+01	1.9351E+01	1.1295E+00	**3.2940E+00**	6.5551E+00	5.1267E+00	2.9285E+00
	p-Value	8.0265E-11	7.5569E-10	7.5569E-10	7.5484E-10	7.5569E-10	7.4113E-10	9.5875E-02	7.5569E-10	**N/A**	7.5569E-10	7.5569E-10	3.7099E-02
	Rank	12.0	7.0	5.0	9.0	4.0	10.0	11.0	6.0	**1.0**	8.0	3.0	2.0
f_{25}	Avg.Obj	2.7698E+03	2.7049E+03	2.7058E+03	2.7054E+03	2.7242E+03	2.7361E+03	2.7666E+03	2.7048E+03	**2.7027E+03**	2.7062E+03	2.7216E+03	2.7161E+03
	Std. dev.	1.1994E+01	2.4505E+00	2.5171E+00	1.4246E+00	1.4781E+01	4.8677E+00	1.1983E+01	1.0253E+00	**3.7170E+00**	1.9780E+00	1.5410E+01	2.3908E+00
	p-Value	5.0194E-11	7.5569E-10	7.5569E-10	7.5569E-10	7.3544E-10	7.2929E-10	2.8083E-11	7.2345E-08	**N/A**	7.5569E-10	6.8144E-10	7.3377E-10
	Rank	12.0	3.0	5.0	4.0	9.0	10.0	11.0	2.0	**1.0**	6.0	8.0	7.0
f_{26}	Avg.Obj	2.7181E+03	2.7084E+03	2.7185E+03	2.7224E+03	2.7005E+03	2.7801E+03	2.7171E+03	2.7084E+03	2.7067E+03	2.7104E+03	2.7004E+03	**2.7003E+03**
	Std. dev.	5.9360E+00	2.7304E+01	3.3277E+01	4.1849E+01	1.2530E-01	4.0301E+01	4.9389E+00	2.7310E+01	1.3534E+01	3.0311E+01	1.2952E-01	**3.0943E-02**
	p-Value	2.4540E-10	1.8586E-05	2.9010E-07	1.6881E-09	7.5569E-10	1.8607E-08	4.5325E-05	1.8586E-05	1.4172E-08	2.1913E-06	7.5569E-10	**N/A**
	Rank	9.0	6.0	10.0	11.0	3.0	12.0	8.0	5.0	4.0	7.0	2.0	**1.0**
f_{27}	Avg.Obj	3.7302E+03	3.1164E+03	3.2908E+03	3.2601E+03	3.6694E+03	3.8203E+03	3.7152E+03	3.1039E+03	3.1997E+03	3.2406E+03	3.6455E+03	**3.1015E+03**
	Std. dev.	9.3888E+01	1.0523E+01	2.9260E+02	1.0986E+02	3.8912E+02	2.2162E+02	1.0250E+02	1.2863E+00	2.0826E+02	1.2823E+02	4.2561E+02	**5.2268E-01**
	p-Value	2.9643E-09	7.5569E-10	1.8648E-09	7.4719E-10	1.5171E-09	9.8073E-09	1.2659E-08	9.2946E-05	7.5569E-10	7.5569E-10	9.5875E-08	**N/A**
	Rank	11.0	3.0	7.0	6.0	9.0	12.0	10.0	2.0	4.0	5.0	8.0	**1.0**
f_{28}	Avg.Obj	7.6623E+03	4.2411E+03	7.0162E+03	3.7952E+03	4.9296E+03	7.5656E+03	7.4073E+03	**3.6514E+03**	5.9942E+03	3.7309E+03	4.9060E+03	5.7270E+03
	Std. dev.	8.2683E+02	3.5222E+02	5.4659E+02	2.0932E+02	6.0352E+02	6.8329E+02	9.2015E+02	**4.2063E+01**	8.1094E+02	1.7864E+02	4.9021E+02	1.9547E+02
	p-Value	1.9425E-09	7.5569E-10	9.1375E-08	7.5569E-10	7.5569E-10	4.5438E-11	3.7190E-11	**N/A**	1.0872E-09	7.5569E-10	7.5569E-10	6.2597E-10
	Rank	12.0	4.0	9.0	3.0	6.0	11.0	10.0	**1.0**	8.0	2.0	5.0	7.0

Table 6 Composition functions: comparison of the CSBO with other optimization algorithms for 30-D problems. The best Avg.Obj results among the 12 algorithms are shown in bold. (cont.)

Method		CSA	FA	KH	MVO	WOA	SBO	CCSA	CFA	CKH	CMVO	CWOA	CSBO
30-D													
f_{29}	Avg.Obj	1.8760E+07	1.6077E+04	7.5539E+06	7.4381E+05	4.4412E+06	2.6289E+06	1.8620E+07	6.2988E+03	1.5826E+06	2.0352E+05	4.2436E+06	**4.2314E+03**
	Std. dev.	1.3167E+07	3.4732E+03	1.2094E+07	2.4760E+06	4.9197E+06	4.5078E+06	1.6577E+07	7.1446E+02	5.4582E+06	1.3232E+06	4.8302E+06	**2.3206E+02**
	p-Value	7.2792E-09	7.5569E-10	1.3831E-09	7.5399E-10	7.4990E-10	7.4029E-10	6.9928E-08	7.5569E-10	7.5569E-10	7.5569E-10	7.5279E-10	**N/A**
	Rank	12.0	3.0	10.0	5.0	9.0	7.0	11.0	2.0	6.0	4.0	8.0	**1.0**
f_{30}	Avg.Obj	1.5790E+06	2.0114E+04	4.5515E+05	9.8550E+03	9.1064E+04	6.2358E+03	1.4505E+06	5.6485E+03	6.0181E+04	9.6229E+03	9.0257E+04	**5.3403E+03**
	Std. dev.	3.8527E+05	6.6873E+03	2.6586E+05	3.7609E+03	7.3361E+04	6.1930E+02	4.5848E+05	8.7147E+02	2.9296E+04	2.3675E+03	6.1458E+04	**5.4975E+02**
	p-Value	1.1101E-09	7.5569E-10	1.2659E-08	7.5569E-10	7.5569E-10	7.0806E-10	3.3668E-09	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**
	Rank	12.0	6.0	10.0	5.0	9.0	3.0	11.0	2.0	7.0	4.0	8.0	**1.0**
Overall rank		92.0	40.0	58.0	50.0	59.0	69.5	83.0	24.5	32.0	42.0	51.0	**23.0**

Table 7 Composition functions: comparison of the CSBO with other optimization algorithms for 50-D problems. The best Avg.Obj results among the 12 algorithms are shown in bold.

Method		CSA	FA	KH	MVO	WOA	SBO	CCSA	CFA	CKH	CMVO	CWOA	CSBO
50-D													
f_{23}	Avg.Obj	4.1389E+03	2.7075E+03	2.7520E+03	2.6504E+03	2.6854E+03	2.6441E+03	4.0451E+03	2.6441E+03	**2.5827E+03**	2.6485E+03	2.6839E+03	2.6440E+03
	Std. dev.	1.7899E+02	1.5485E+01	9.8725E+01	3.0814E+00	2.9707E+01	7.3895E-03	2.4907E+02	1.0667E-02	**7.4011E+01**	1.8486E+00	1.0428E+01	5.8747E-04
	p-Value	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**	7.5569E-10	7.5569E-10	7.5569E-01
	Rank	12.0	9.0	10.0	6.0	8.0	3.5	11.0	3.5	**1.0**	5.0	7.0	2.0
f_{24}	Avg.Obj	3.0495E+03	2.6725E+03	2.6421E+03	2.6799E+03	**2.6003E+03**	2.7028E+03	3.0440E+03	2.6721E+03	2.6070E+03	2.6791E+03	**2.6003E+03**	2.6367E+03
	Std. dev.	4.0193E+01	2.6671E+00	2.4170E+01	3.3146E+00	**5.4160E-01**	2.6180E+01	3.4171E+01	5.1024E+00	1.4281E+01	3.6231E+00	**5.5893E-01**	2.1385E+01
	p-Value	7.5569E-10	7.5569E-10	7.5569E-10	7.5228E-10	**N/A**	7.5484E-10	7.5569E-10	7.5569E-10	1.8153E-03	7.5569E-10	**N/A**	2.1636E-03
	Rank	12.0	7.0	5.0	9.0	**1.5**	10.0	11.0	6.0	3.0	8.0	**1.5**	4.0
f_{25}	Avg.Obj	2.8819E+03	2.7119E+03	2.7104E+03	2.7314E+03	2.7168E+03	2.7634E+03	2.8797E+03	2.7065E+03	**2.7003E+03**	2.7149E+03	2.7159E+03	2.7314E+03
	Std. dev.	2.3802E+01	2.3259E+00	4.0475E+00	3.1715E+00	3.2142E+01	8.7552E+00	2.9474E+01	6.5297E+00	**2.2481E+00**	3.7768E+00	2.8957E+01	1.5111E-01
	p-Value	6.0844E-11	7.5569E-10	7.5569E-10	3.8094E-12	7.5036E-11	7.5313E-10	4.7989E-07	7.5569E-10	**N/A**	7.5569E-10	9.1357E-11	2.1636E-10
	Rank	12.0	4.0	3.0	8.5	7.0	10.0	11.0	2.0	**1.0**	5.0	6.0	8.5
f_{26}	Avg.Obj	2.7311E+03	2.7943E+03	2.7975E+03	2.7887E+03	**2.7005E+03**	2.7982E+03	2.7270E+03	2.7846E+03	2.7967E+03	2.7711E+03	**2.7005E+03**	2.7078E+03
	Std. dev.	1.4777E+01	2.3406E+01	1.8168E+01	8.4199E+01	**1.0711E-01**	1.4108E+01	1.0922E+01	3.6026E+01	1.7814E+01	9.1783E+01	**9.1350E-02**	2.1179E+01
	p-Value	7.5569E-10	1.0872E-09	9.0681E-10	7.5569E-10	**N/A**	7.9968E-10	1.6772E-03	6.3812E-09	9.0681E-10	5.0608E-04	**N/A**	1.0177E-01
	Rank	5.0	9.0	11.0	8.0	**1.5**	12.0	4.0	7.0	10.0	6.0	**1.5**	3.0
f_{27}	Avg.Obj	5.2136E+03	3.7932E+03	5.1924E+03	3.6682E+03	4.8355E+03	4.7212E+03	5.1939E+03	3.6832E+03	4.3665E+03	3.6561E+03	4.8055E+03	**3.4928E+03**
	Std. dev.	5.7171E+01	1.2750E+02	3.1406E+02	1.0880E+02	1.0442E+02	2.1927E+02	8.6634E+01	1.9267E+02	1.5245E+02	1.1728E+02	1.3498E+02	**4.8228E+02**
	p-Value	4.0620E-11	7.5569E-10	9.9615E-11	7.5228E-10	7.5569E-10	7.5399E-10	2.5435E-11	7.4382E-10	7.5569E-10	6.5295E-05	7.5569E-10	**N/A**
	Rank	12.0	5.0	10.0	3.0	9.0	7.0	11.0	4.0	6.0	2.0	8.0	1.0

Table 7 Composition functions: comparison of the CSBO with other optimization algorithms for 50-D problems. The best Avg.Obj results among the 12 algorithms are shown in bold. (cont.)

Method		CSA	FA	KH	MVO	WOA	SBO	CCSA	CFA	CKH	CMVO	CWOA	CSBO
50-D													
f_{28}	Avg.Obj	1.4253E+04	7.1208E+03	1.1834E+04	4.5947E+03	7.8565E+03	1.2274E+04	1.4175E+04	**4.0265E+03**	9.3555E+03	4.5181E+03	7.6341E+03	8.6528E+03
	Std. dev.	8.3188E+02	6.6624E+02	1.4449E+03	4.1304E+02	1.5909E+03	1.2712E+03	9.6319E+02	**8.0207E+01**	1.0213E+03	4.9555E+02	1.6450E+03	7.7053E+02
	p-Value	7.5569E-10	7.5569E-10	8.5342E-10	7.3878E-10	7.5569E-10	1.8648E-09	2.6731E-11	**N/A**	7.5569E-10	7.5569E-10	7.5569E-10	1.5375E-09
	Rank	12.0	4.0	9.0	3.0	6.0	10.0	11.0	**1.0**	8.0	2.0	5.0	7.0
f_{29}	Avg.Obj	7.9630E+07	2.8216E+04	3.4356E+08	2.7550E+06	2.9042E+07	3.7709E+06	6.0688E+07	1.0337E+04	3.1187E+07	9.3730E+05	2.4111E+07	**8.7044E+03**
	Std. dev.	5.1018E+07	4.4022E+03	2.8723E+08	1.0794E+07	2.3261E+07	1.2994E+07	3.8746E+07	1.1031E+03	1.4731E+08	6.1786E+06	2.0896E+07	**1.0284E+03**
	p-Value	4.9524E-11	7.5569E-10	2.7420E-11	7.5569E-10	5.6855E-09	7.5535E-10	6.2591E-09	7.5569E-10	3.7260E-07	7.5569E-10	7.5552E-10	**N/A**
	Rank	11.0	3.0	12.0	5.0	8.0	6.0	10.0	2.0	9.0	4.0	7.0	**1.0**
f_{30}	Avg.Obj	1.3594E+07	1.5207E+05	2.4083E+06	3.5503E+04	1.0525E+05	1.7875E+04	1.2856E+07	1.4289E+05	1.0046E+05	3.2395E+04	9.0749E+04	**1.3086E+04**
	Std. dev.	3.0940E+06	5.8782E+04	1.5522E+06	1.1145E+04	6.1549E+04	2.4934E+03	3.5440E+06	5.0709E+04	4.7341E+04	8.5506E+03	4.9049E+04	2.5788E+02
	p-Value	6.1064E-09	7.5569E-10	7.5569E-10	7.5569E-10	7.5569E-10	7.5399E-10	1.1471E-10	7.5535E-10	7.5569E-10	7.5569E-10	7.5569E-10	**N/A**
	Rank	12.0	9.0	10.0	4.0	7.0	2.0	11.0	8.0	6.0	3.0	5.0	**1.0**
Overall rank		88.0	50.0	70.0	46.5	48.0	60.5	80.0	33.5	44.0	35.0	41.0	**27.5**

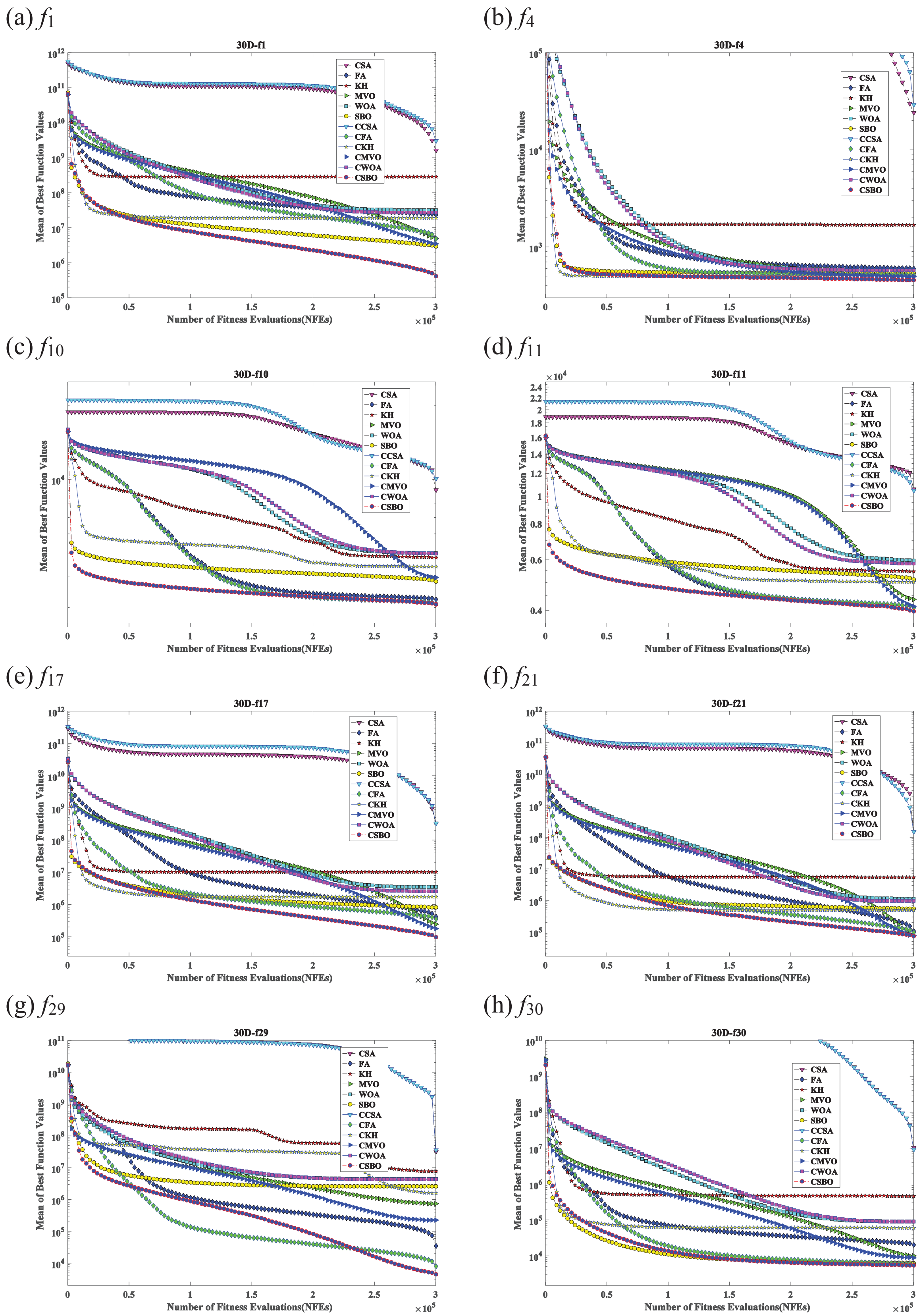


Figure 1 Convergence performance of all 12 optimization algorithms in eight benchmark functions ($f_1, f_4, f_{10}, f_{11}, f_{17}, f_{21}, f_{29}, f_{30}$) at 30-D.

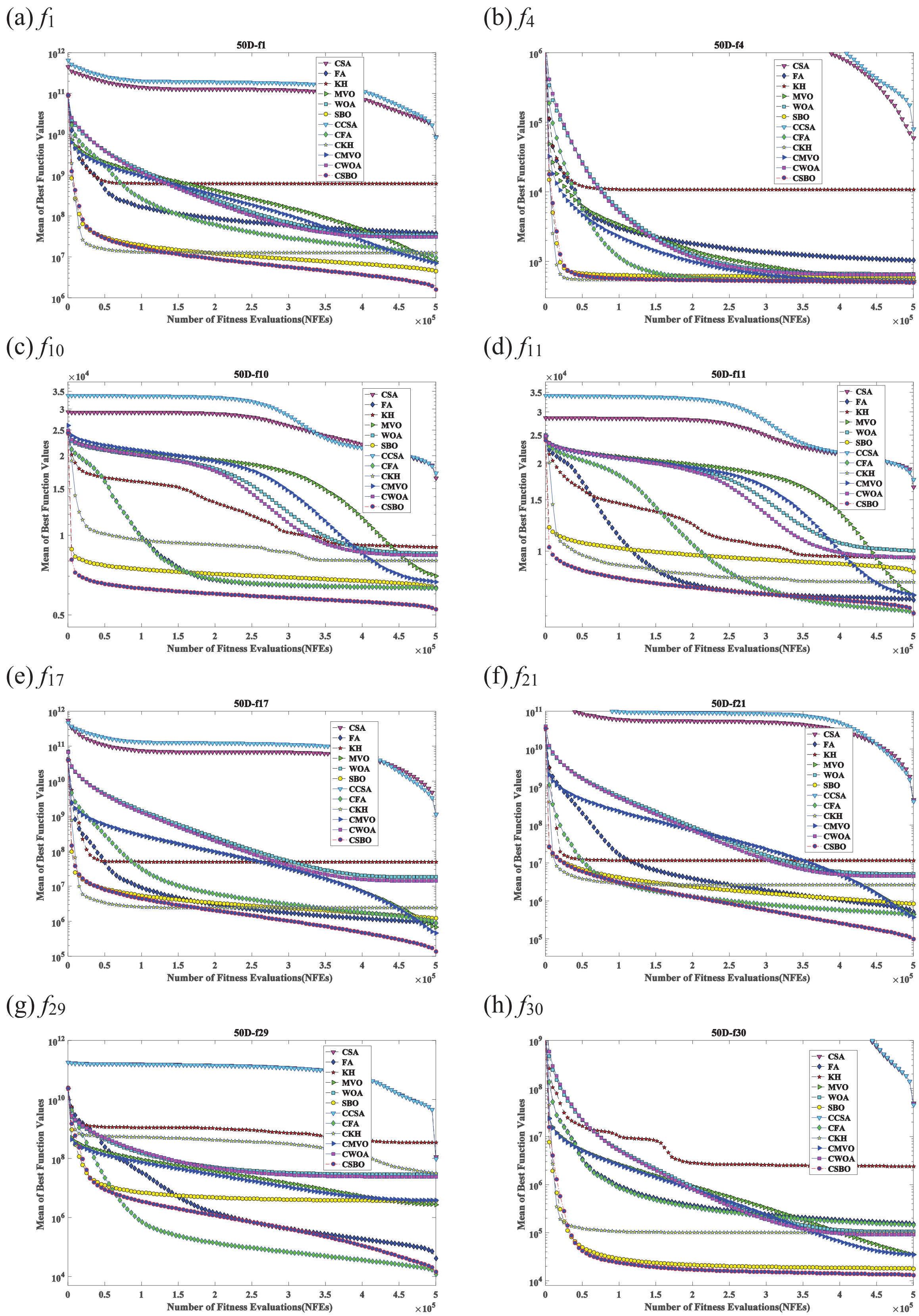
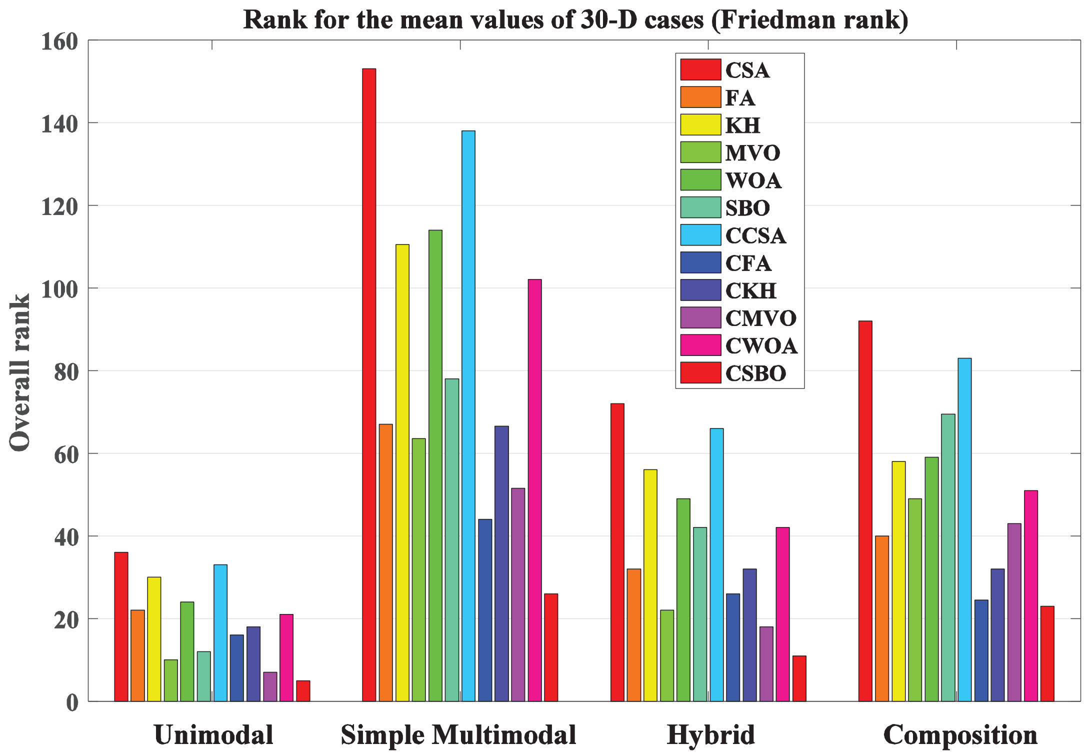


Figure 2 Convergence performance of all 12 optimization algorithms in eight benchmark functions ($f_1, f_4, f_{10}, f_{11}, f_{17}, f_{21}, f_{29}, f_{30}$) at 50-D.

(a)



(b)

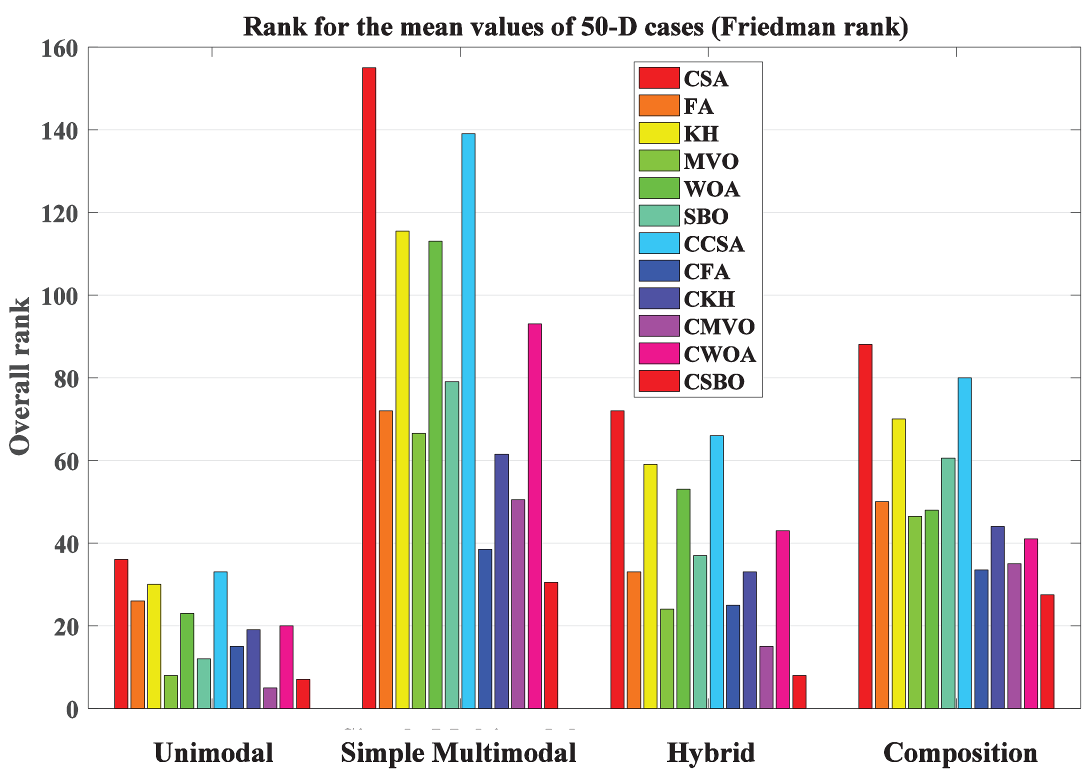


Figure 3 Rank for the mean values of 30-D and 50-D cases (Friedman rank).

Conclusions and future scope

We present a novel, improved meta-heuristic SBO using a wide variety of chaotic maps (CSBO), in which to tune the main parameter, the greatest step size (α) of the standard SBO ; in order to solve complex optimization problems. The numerical results of the experiment show that this novel Tent map chaotic algorithm can greatly enhance the performance of the basic SBO. Moreover, the tuned SBO significantly enhanced the reliability of the global optimality and the quality of the solutions (at CEC2014) of the newly formed algorithm, due to the application of deterministic chaotic signals in place of constant values. In order to evaluate the algorithm with its original (SBO) and improved (CSBO) variants, other mathematical benchmark examples were employed. The statistical results and success rates of the CSBO suggest that the tuned algorithms clearly improve the reliability of the global optimality, and further enhance the quality of the results.

The CSBO proved to be simple and easy to implement within all of our applied (and similar type) applications. In future works, we intend to investigate the further capabilities of the CSBO algorithm in solving real-world engineering problems, as well as discrete optimization problems.

References

Askarzadeh, A. (2016). A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm, *169*, 1–12.

Chintam, J., & Daniel, M. (2018). Real-Power Rescheduling of Generators for Congestion Management Using a Novel Satin Bowerbird Optimization Algorithm. *Energies*, *11*(1), 183.

Frank Wilcoxon. (1945). Individual comparisons by ranking methods. *Biometrics Bulletin*, *1*(6), 80–83.

Gandomi, A.H., Yang, X.S., Talatahari, S., & Alavi, A.H. (2013). Firefly algorithm with chaos. *Communications in Nonlinear Science and Numerical Simulation*, *18*(1), 89–98.

Hatamlou, A., Abdullah, S., & Hatamlou, M. (2011). Data Clustering Using Big Bang-Big Crunch Algorithm. *Innovative Computing Technology*, *241*, 383–388. https://doi.org/10.1007/978-3-642-27337-7_36

Hossein, A. (2012). Krill herd: A new bio-inspired optimization algorithm, *17*, 4831–4845.

Huang, L., Ding, S., Yu, S., Wang, J., & Lu, K. (2015). Chaos-enhanced Cuckoo search optimization algorithms for global optimization. *Applied Mathematical Modelling*, *40*(5–6), 3860–3875.

Kaur, G., & Arora, S. (2018). Chaotic Whale Optimization Algorithm. *Journal of Computational Design and Engineering*, *5*(3), 275–284.

Liang, J., Qu, B., & Suganthan, P. (2014). *Problem Definitions and Evaluation Criteria for the CEC Special Session and Competition on Single Objective Real-parameter Numerical Optimization, Technical Report 201311,*. Singapore, 2014, pp.: Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China and Nanyang Technological University,.

Lorenz, E.N. (1963). Deterministic Nonperiodic Flow. *Journal of Atmospheric Sciences.*, *20 (2)*, 130–148.

Mirjalili, S., & Gandomi, A.H. (2017). Chaotic gravitational constants for the gravitational search algorithm. *Applied Soft Computing Journal*, *53*, 407–419.

Mirjalili, S., & Lewis, A. (2016). The Whale Optimization Algorithm. *Advances in Engineering Software*, *95*, 51–67. https://doi.org/10.1016/j.advengsoft.2016.01.008

Mirjalili, S., Mirjalili, S.M., & Hatamlou, A. (2016). Multi-Verse Optimizer : a nature-inspired algorithm for global optimization. *Neural Computing and Applications*, (27), 495–513. https://doi.org/10.1007/s00521-015-1870-7

Problems, O., Hassanien, A.E., Bhattacharyya, S., & Hassanien, A.E. (2018). Chaotic Crow Search Algorithm for Fractional Optimization Problems. *Applied Soft Computing*.

Samareh Moosavi, S.H., & Khatibi Bardsiri, V. (2017). Satin bowerbird optimizer: A new optimization algorithm to optimize ANFIS for software development effort estimation. *Engineering Applications of Artificial Intelligence*, *60*, 1–15.

Sayed, G.I., Khoriba, G., & Haggag, M.H. (2018). A novel chaotic salp swarm algorithm for global optimization and feature selection. *Applied Intelligence*, *48*(10), 3462–3481.

Wang, G.G., Guo, L., Gandomi, A. H., Hao, G.S., & Wang, H. (2014). Chaotic Krill Herd algorithm. *Information Sciences*, *274*, 17–34.

Wangchamhan, T., Chiewchanwattana, S., & Sunat, K. (2017). Efficient algorithms based on the k-means and Chaotic League Championship Algorithm for numeric, categorical, and mixed-type data clustering. *Expert Systems with Applications*, *90*, 146–167.

Xu, Q., Wang, S., Zhang, L., & Liang, Y. (2013). A novel chaos danger model immune algorithm. *Communications in Nonlinear Science and Numerical Simulation*, *18*(11), 3046–3060. https://doi.org/10.1016/j.cnsns.2013.04.017

Yang, D., Liu, Z., & Zhou, J. (2014). Chaos optimization algorithms based on chaotic maps with different probability distribution and search speed for global optimization. *Commun Nonlinear Sci Numer Simulat*, *19*, 1229–1246.

Yang, X.-S. (2010). Firefly algorithm, L´evy flights and global optimization. *In: Research and Development in Intelligent Systems XXVI (Eds M. Bramer, R. Ellis, M. Petridis), Springer London,* 209–218.

Yuan, X., Zhang, T., Xiang, Y., & Dai, X. (2015). Parallel chaos optimization algorithm with migration and merging operation. *Applied Soft Computing Journal*, *35*, 591–604.

Yuan, X., Zhao, J., Yang, Y., & Wang, Y. (2014). Hybrid parallel chaos optimization algorithm with harmony search algorithm. *Applied Soft Computing*, *17*, 12–22.