International Journal of Engineering & Technology, 7 (3.15) (2018) 73-79

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Chaotic Immune Symbiotic Organisms Search Algorithm for Solving Optimisation Problem

Mohamad Khairuzzaman Mohamad Zamani[1]*, Ismail Musirin[2,#], Saiful Izwan Suliman[3], Sharifah Azma Syed Mustaffa[4]

[1,2]Power System Operation Computational Intelligence Research Group (POSC), Centre for Electrical Power Engineering Studies (CEPES), Faculty of Electrical Engineering, Universiti Teknologi MARA, 40450 Shah Alam, Selangor, Malaysia
[3] Faculty of Electrical Engineering, Universiti Teknologi MARA, 40450 Shah Alam, Selangor, Malaysia
[4] College of Engineering, Universiti Tenaga Nasional, Kajang, Selangor, Malaysia
**Corresponding author E-mail: mohd_khairuzzaman@yahoo.com*
[#]Corresponding author E-mail: ismailbm1@gmail.com

Abstract

Achieving an optimal solution is very crucial while solving a problem. To achieve the optimality required, optimisation techniques can be implemented while solving the problem. The presence of classical optimisation techniques has enabled an optimal solution to be obtained. However, as the complexity of the optimisation problem increased, classical optimisation techniques faced difficulties in providing optimal solutions. Heuristics-based algorithms were introduced to counter the problem faced by classical optimisation techniques. Good performance of these heuristics-based algorithm has been implied through various implementation in solving optimisation problems. Despite the performance of these algorithms, the flaws of these algorithms hinder them from producing high-quality results. To mitigate the problem, this paper presents the development of Chaotic Immune Symbiotic Organisms Search algorithm which was inspired by the element of diversification as well as the increased capability of exploration. The performance of the proposed algorithm has been tested by solving several benchmark test functions. A comparative study was also conducted with respect to several other existing optimisation algorithms resulted in the superiority of the proposed algorithm in providing high-quality solutions.

Keywords: *Benchmark Test Functions; Chaotic Immune Symbiotic Organisms Search; Chaotic Local Search*

1. Introduction

While solving a problem, it is important that the solution was chosen for the problem to be optimal. In the past, problems are solved through experimentation and hit-and-miss approach. However, the presence of optimisation techniques has changed the scenario where an algorithm is implemented to solve a problem mathematically, which then replaces the hit-and-miss approach. In early days, classical optimisation techniques are implemented to solve optimisation problems such as Lagrange Multipliers [1], Direct Search [2], Random Search Algorithm [3], Linear Programming [4] and Non-Linear Programming [5]. However, these classical optimisation techniques suffer problems such as convergence to local optima solutions, unsuitability of classical optimisation techniques to solve non-convex and discontinuous functions, as well as the solution quality produced by the classical optimisation techniques, are highly dependent to its starting point [6][7][8]. Authors in [8] also reported that discontinuous nature of the optimisation problem could affect the performance of NLP while Dynamic Programming (DP) technique can suffer from a serious problem related to the dimensionality of the optimisation problem.

Heuristics-based optimisation algorithms are introduced to overcome the drawbacks of the classical optimisation techniques. These heuristics-based algorithms have been developed and widely implemented to solve various optimisation problems. Genetic Algorithm (GA) is an algorithm listed under Evolutionary Algorithm (EA) family has been implemented to solve various optimisation problems such as optimal machining parameter for metal cutting process [9], optimisation of wind turbines blade [10] as well as optimisation of optical parameter of multilayer bandpass filter [11]. Another algorithm listed under EA known as Evolutionary Programming (EP) was also implemented to solve optimisation of heating, ventilation and air-conditioning (HVAC) system [12], economic dispatch of power generation in power system [13], as well as minimisation of additional chain length, reduce the computational requirement in cryptosystem applications [14]. A metaheuristic optimisation algorithm known as Particle Swarm Optimization (PSO) has also been widely used to solve problems such as power consumption minimisation for luminance control [15], optimal design of truss structures [16] and overload management of power transmission system [17]. Many other optimisation algorithms have been developed and implemented to solve various optimisation problems such as Artificial Immune System (AIS) [18], Ant Colony Optimization (ACO) [19], Artificial Bee Colony (ABC) algorithm [20], Firefly Algorithm (FA) [21], Gravitational Search Algorithm (GSA) [22] as well as many other optimisation techniques available.

Despite the development of these optimisation algorithms, some algorithms suffer drawback from producing a high-quality optimal solution. At the same time, the nature of metaheuristic algorithms in which it does not promise a globally-optimal solution to be found worsening the problem. Throughout the implementation of optimisation algorithms, it has been reported that GA suffers from high computation time [23]. Authors in [24] also revealed that GA tends to produce low-quality optimal solution when dealing with large-scale optimisation problem. While PSO is said to provide faster convergence and computationally inexpensive, it also suffers from tendency to stuck at local optima point [23][25]. Reference [25] also stated that the performance of PSO algorithm is very dependent on the optimisation parameters. In [7], FA was reported to produce suboptimal solutions as well as the tendency not to be able to converge if improper selection of its parameters is chosen. The same problem was also reported in [26] for Harmony Search (HS) algorithm where improper parameter choice can hinder HS performance.

Doddy and Prayogo have introduced an optimisation algorithm known as Symbiotic Organisms Search (SOS) in 2014. This algorithm simulates the symbiotic relationship of organisms living in an ecosystem. SOS boast its superiority from other optimisation algorithms by having no control parameters that govern the behaviour of the algorithm, which eliminates the problem of an algorithm for being dependent on controlling parameter of an algorithm to produce a high-quality optimal solution [27]. The proposed SOS algorithm has been adopted to solve various optimisation problems such as voltage profile improvement in a power system [28], optimal power flow [29] as well as economic dispatch problems [30]. Further research has been conducted to improve the search capability of SOS algorithm through introduction of Chaotic Local Search (CLS) algorithm to the novel SOS algorithm termed as Chaos Embedded Symbiotic Organisms Search (CSOS), resulting in increased performance of the modified algorithm [31]. Despite the improvement made to the algorithm, it is beneficial that the optimisation algorithm to be improved further in the attempt to produce a higher-quality optimal solution.

This paper presents the development of a hybrid optimisation algorithm termed as Chaotic Immune Symbiotic Organisms Search (CISOS) algorithm for solving optimisation problems. The proposed CISOS algorithm was inspired by the novel SOS algorithm and the improvement made in [31]. From the algorithm developed in [31], the potentially best solution is updated by choosing the best organism from the ecosystem as well as the organism found by CLS. In CISOS, Cloning and Elitism Tournament which were inspired from AIS algorithm was embedded in the modified algorithm as in [31]. The inclusion of these elements suggested that the organisms in the ecosystem can be cloned to produce more duplicates and then compete among themselves in order to maintain its survivability in the ecosystem. The cloning of the organisms will increase the number of possible solutions while the tournament will ensure that the organisms with the highest solution quality will be chosen to be carried forward to the next iteration of the optimisation process. Section 2 of this paper will discuss the optimisation process of the algorithm. The performance of the algorithm will be presented in Section 3 of the paper while the paper is concluded in Section 4.

2. Chaotic Immune Symbiotic Organisms Search (CISOS)

In this paper, a hybrid heuristic algorithm termed as Chaotic Immune Symbiotic Organisms Search (CISOS) has been proposed and developed. The proposed algorithm was inspired by the novel SOS algorithm which has been first proposed by Cheng and Prayogo. SOS emulates the symbiosis relationship of organisms in an ecosystem. While SOS has proved to have a good performance, there is a need to further improve the algorithm so that it can provide higher quality solutions. To achieve this, element of cloning and elitism competition which were inspired from AIS was introduced where cloning of the organisms will diversify the organisms in the ecosystem, hence increasing the number of possible solutions which would be yielded by the optimisation process. The element of competition will ensure that only high-quality solutions will be carried forward to the next iteration of the optimisation algorithm and eliminating the lower quality solutions. The search capability of the proposed algorithm was boosted through the incorporation of CLS in the algorithm. The addition of CLS will assist the algorithm to escape from being stuck at the local optima point, hence increase the capability of the algorithm to provide a higher quality solution. The pseudocode of the proposed algorithm is illustrated in Figure 1, and the flow of the algorithm is discussed as follows.

Step 1 : **Initialisation.** During initialisation, a set of possible solutions known as the organism is generated. The organism is generated randomly and bounded within its minimum and maximum bound. The generation is done for N_{org} times where N_{org} denotes the total number of initial organisms generated during initialisation. After N_{org} organisms have been generated, the organism with the best fitness value is chosen from the pool of organisms termed as an ecosystem. Then, the chaotic search space radius is determined as in (2)

$$x_{i,j} = rand\left(x_{j,\min}, x_{j,\max}\right) \tag{1}$$

$$r_j = \frac{x_j^{\max} - x_j^{\min}}{2} \tag{2}$$

Step 2 : **Cloning.** In this phase, each organism will be duplicated by N_{clone} copies where N_{clone} is the number of clones for each organism. By cloning the organisms, it allows the optimisation algorithm to have more possible solutions. After all organisms has been cloned, then the ecosystem will have N_{Torg} organisms. N_{Torg} is defined as the total number of organisms in the ecosystem after cloning process. N_{Torg} can be expressed as:

$$N_{Torg} = N_{org} \times N_{clone} \tag{3}$$

After the organisms has been cloned, the organisms counter *i* is initialised to 1.

Step 3 : **Mutualism Phase.** At this stage, the organisms are subjected to mutualistic symbiosis relationship. A j^{th} organism is chosen from the ecosystem in which that i^{th} organism and jth organism are not the same. Through mutualistic relationship, 2 new organisms namely $X_{i,new}$ and $X_{j,new}$ are produced. Then, the fitness value of $X_{i,new}$ and $X_{j,new}$ are compared with the fitness value of X_i and X_j respectively. The organisms with a better fitness value will be included in the ecosystem while the worse ones are eliminated. The mutualistic relationship of the organisms can be mathematically represented as in (4) and (5).

$$X_i \neq X_j \tag{4}$$

$$X_{i,new} = X_i + rand(0,1) \times \left(X_{best} - \left(MV \times BF1\right)\right) \tag{5}$$

$$X_{j,new} = X_j + rand(0,1) \times \left(X_{best} - \left(MV \times BF2\right)\right) \tag{6}$$

$$MV = \frac{X_i + X_j}{2} \tag{7}$$

where *rand(0,1)* is a random number between 0 and 1, *MV* is the mutual vector; the best organism is denoted as X_{best}. *BF1* and *BF2* are the benefit factor, and it is a random integer number either 1 or 2.

Step 4 : **Commensalism Phase.** In this phase, the organisms are subjected to commensal symbiosis relationship. An organism X_j known as j^{th} organism is chosen among the organisms in the ecosystem. A new organism $X_{i,new}$ is then produced from the commensal relationship. The fitness value of $X_{i,new}$ is evaluated and compared with the fitness value of X_i. If $X_{i,new}$ possess a better fitness value compared to the fitness value of X_i, then X_i,new will replace X_i in the ecosystem. Otherwise, X_i is retained, and $X_{i,new}$ is eliminated. The commensal relationship of the organisms can be expressed mathematically as in (9). In (9), *rand(-1,1)* is defined as a random number ranged between -1 to 1.

$$X_i \neq X_j \tag{8}$$

$$X_{i,new} = X_i + rand\left(-1,1\right) \times \left(X_{best} - X_j\right) \tag{9}$$

Step 5 : **Parasitism Phase.** In this phase, the organisms are subjected to parasitic symbiosis relationship. An organism X_j is chosen among the organisms in the ecosystem in which X_i and X_j are not the same organisms. Later, a parasite called *parasite_vector* is produced by duplicating X_i. Then, random dimension of the *parasite_vector* is then modified by using random number bounded by the minimum and maximum limit of the variable with respect to that dimension. In parasitic symbiosis relationship, the parasite will try to replace the host and reside itself in the host place. In this case, X_j is considered to be the host.

The fitness value of the *parasite_vector* is then evaluated and compared with the fitness value of X_j. In the case of *parasite_vector* possessing a better fitness value compared to X_j, then X_j is killed, and *parasite_vector* will replace the host place in the ecosystem. On the other hand, if X_j has a better fitness value compared to *parasite_vector*, then the host is having immunity from the parasite, hence maintains its location in the ecosystem while eliminating the parasite.

$$X_i \neq X_j \tag{10}$$

Step 6 : **Best Organism Identification.** During this stage, the organism with the best fitness value will be considered as the candidate for the best organism in the ecosystem X_{best_new}. Then, the fitness value of X_{best_new} is compared with the fitness value of Xbest. If the fitness value of X_{best_new} is better compared to X_{best}, then X_{best_new} will replace X_{best}. If X_{best} has a better fitness value compared to X_{best_new}, then X_{best} is maintained, and X_{best_new} is discarded.

Step 7 : **Symbiosis Relationship Termination Test.** In this stage, the algorithm will determine if all the organisms have undergone the symbiosis relationship phases. If all organisms have undergone these phases, the algorithm will proceed to step 8. Otherwise, the organism counter *i* is updated and the algorithm continues at step 3.

Step 8 : **Elimination.** At this stage, the size of the ecosystem is shrunk by retaining N_{org} fittest organisms from the ecosystem while eliminating the rests. These fitter organisms will be carried forward to the next iteration of the algorithm.

Step 9 : **Chaotic Local Search.** At this stage, CLS algorithm is implemented in order to improve the searching capability of the algorithm as well as preventing the algorithm from being stuck at a local optima point. The search is initiated by setting the CLS iteration counter *c* to 0, and the chaotic variable cv_c is set using a random number between 0 to 1. The value of cv_c is updated by using a chaotic map known as Piecewise Linear Chaotic Map (PLCM). The updated chaotic variable cv_{c+1} is then used to generate a possible solution $X_{cls,c}$ by mapping cv_{c+1} around the best organism and it is expressed as in (14). The fitness value of $X_{cls,c}$ is evaluated and compared with the fitness value of the best organism. If $X_{cls,c}$ possess a better fitness value than fitness value of X_{best}, then $X_{cls,c}$ will replace Xbest, the local search is terminated, the chaotic search space radius r_j is shrunken using (15), and the optimisation process is continued to step 10. Otherwise, the CLS iteration counter *c* is updated, and the chaotic variable is updated again and continued further until $X_{cls,c}$ achieve better fitness value than X_{best}. In the case of *c* approaches the maximum CLS iteration limit c_{max}, then the local search is terminated, the chaotic search space radius r_j is shrunken using (15), and the algorithm continues to step 10.

$$c = 0 \tag{11}$$

$$cv_c = rand\left(0,1\right) \tag{12}$$

$$cv_{c+1} = \begin{cases} \dfrac{cv_c}{p} & cv_c \in \left(0, p\right) \\ \dfrac{\left(1-cv_c\right)}{\left(1-p\right)} & cv_c \in \left[p, 1\right) \end{cases} \tag{13}$$

$$X_{cls,c} = X_{best} + r_j\left(2cv_{c+1} - 1\right) \tag{14}$$

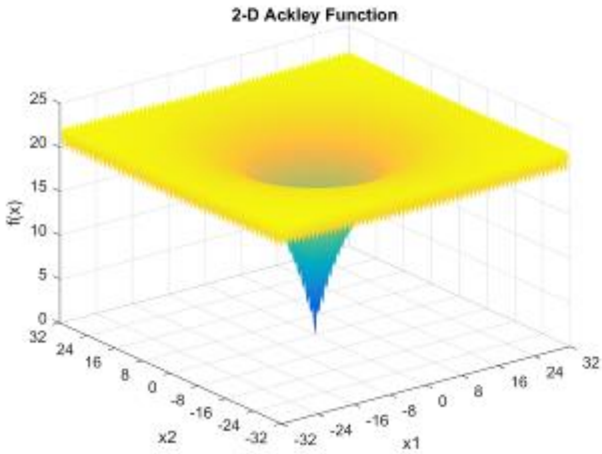
$$r_j = r_j \times rand\left(0,1\right) \tag{15}$$

Step 10 : **Convergence Test.** At this stage, if the algorithm has reached its maximum iteration value, then the optimisation process is stopped, and the best organism is taken as the optimal solution provided by the algorithm. If not, the iteration counter will be updated, and the optimisation process continues to step 2.

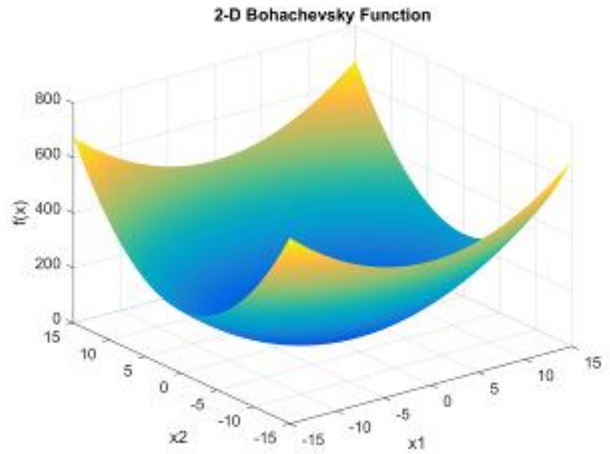
CISOS Algorithm

```
Define objective function
Initialise the ecosystem of N_org organisms using random values
Initialise the best organism X_best in the ecosystem
Set chaotic search space radius
Set CISOS iteration counter t = 0
while t < iter_max
    Clone each of the organism by N_clone clones
    while i <= N_Torg
        Subject the organisms to mutualism phase
        Subject the organisms to commensalism phase
        Subject the organisms to parasitism phase
        Determine and update the best organism X_best
        i = i + 1
    end while
    Retain N_org fittest organisms and eliminate the other organisms
    Set CLS iteration counter, c = 0 and found_better_solution = false
    Initialise chaotic variable cv_c
    do
        Update chaotic variable cv_c+1 using (13)
        Generate X_cls,c by mapping cv_c+1 around X_best using (14)
        Evaluate the fitness value of X_cls,c
        if the fitness value of X_cls,c is better than X_best
            found_better_solution = true
            Update the best organism by replacing X_best with X_cls,c
        end if
        c = c + 1
    while a better solution is not found and c ≤ c_max
    Reduce the chaotic search space radius
t = t + 1
end while
```

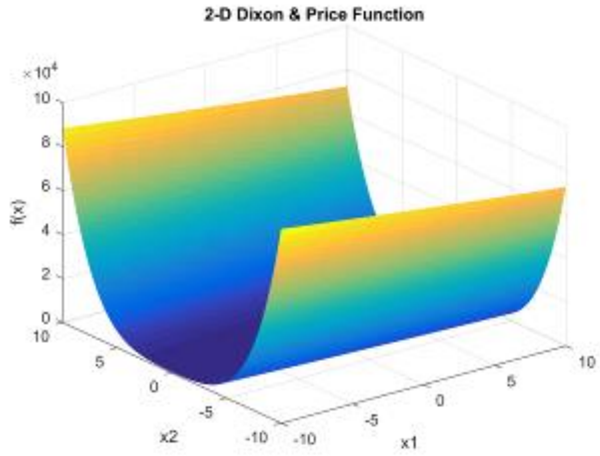
Fig. 1: Pseudocode of the proposed CISOS algorithm


2-D Ackley Function
f(x)
x2
x1
25
20
15
10
5
0
32 24 16 8 0 -8 -16 -24 -32
-32 -24 -16 -8 0 8 16 24 32

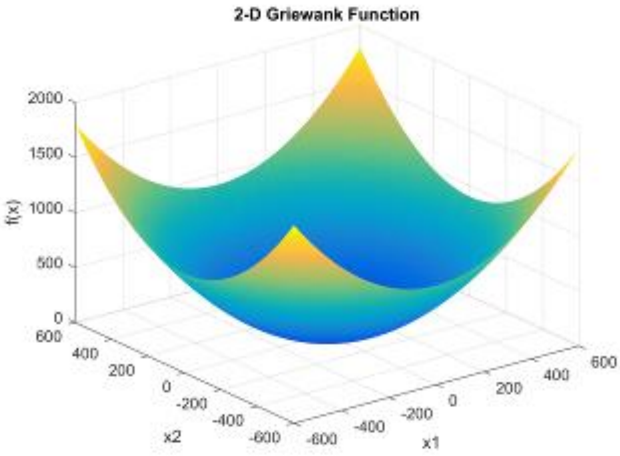
(a)


2-D Bohachevsky Function
f(x)
x2
x1
800
600
400
200
0
15 10 5 0 -5 -10 -15
-15 -10 -5 0 5 10 15

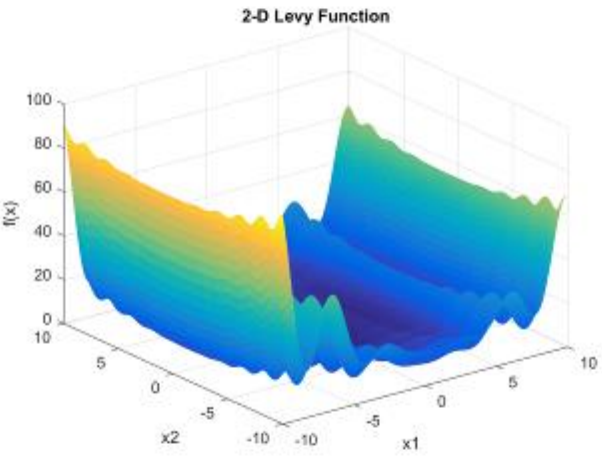
(b)


2-D Dixon & Price Function
×10^4
f(x)
x2
x1
10
8
6
4
2
0
10 5 0 -5 -10
-10 -5 0 5 10

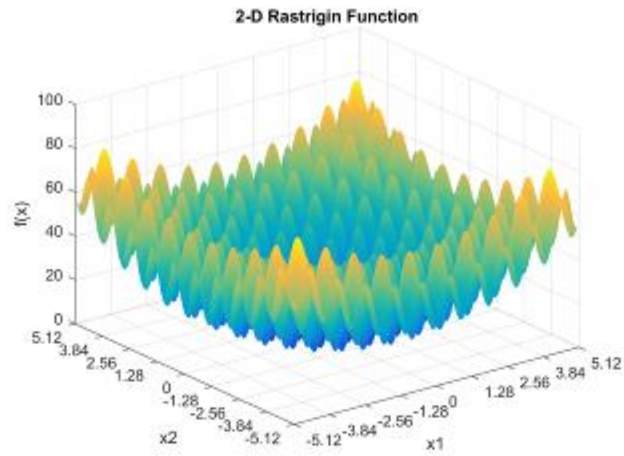
(c)


2-D Griewank Function
f(x)
x2
x1
2000
1500
1000
500
0
600 400 200 0 -200 -400 -600
-600 -400 -200 0 200 400 600

(d)


2-D Levy Function
f(x)
x2
x1
100
80
60
40
20
0
10 5 0 -5 -10
-10 -5 0 5 10

(e)


2-D Rastrigin Function
f(x)
x2
x1
100
80
60
40
20
0
5.12 3.84 2.56 1.28 0 -1.28 -2.56 -3.84 -5.12
-5.12 -3.84 -2.56 -1.28 0 1.28 2.56 3.84 5.12

(f)

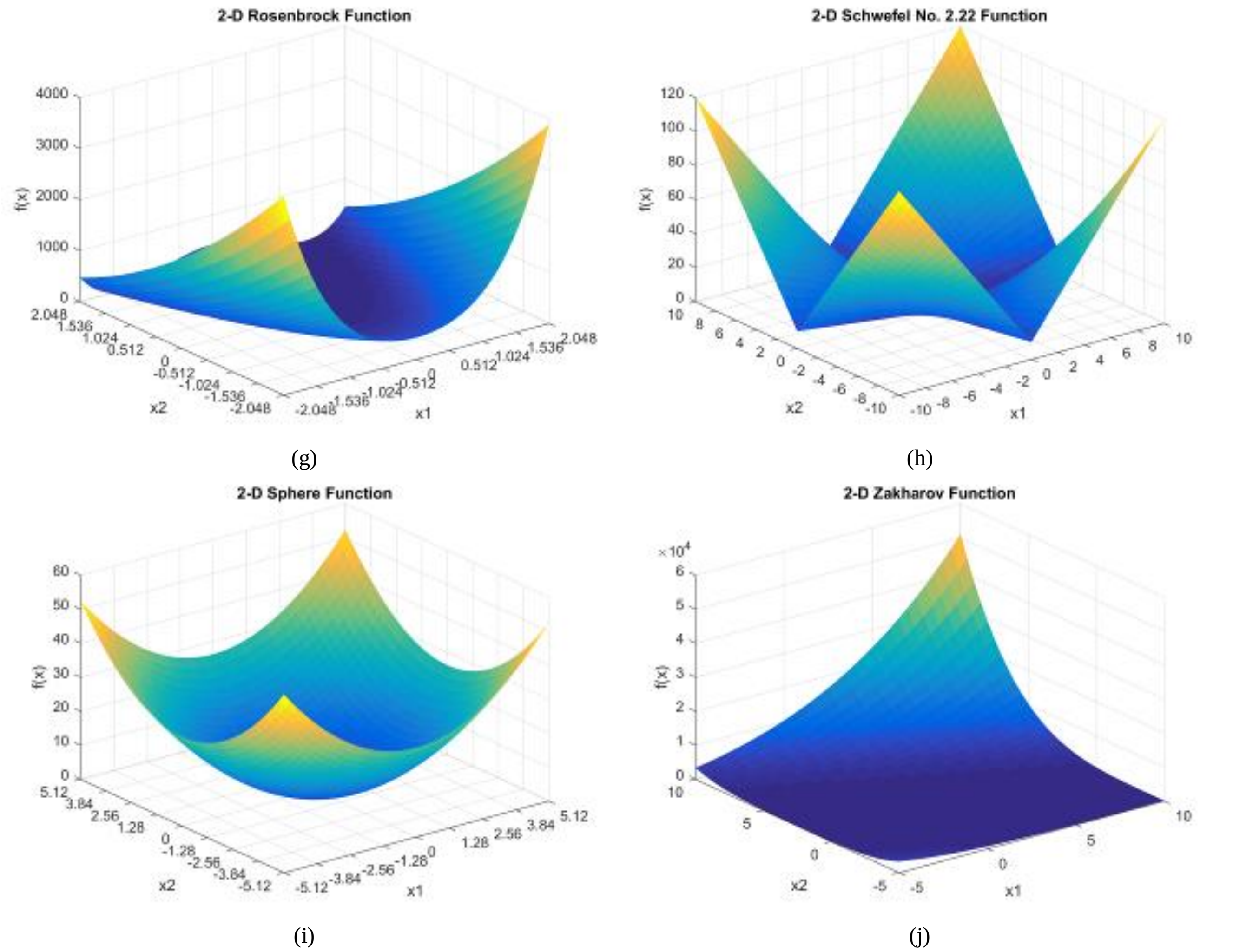


Fig. 2: 2-dimensional plot of benchmark test functions. (a) Ackley (b) Bohachevsky (c) Dixon & Price (d) Griewank (e) Levy (f) Rastrigin (g) Rosenbrock (h) Schwefel No. 2.22 (i) Sphere (j) Zakharov

3. Performance Analysis

The performance of the proposed algorithm is tested by solving 10 benchmark test functions. The dimension number for all test functions are set to 50 dimensions. These test functions can be categorised as unimodal and multimodal functions. While unimodal functions are easier to be solved, multimodal functions pose a challenge to the optimisation algorithm due to the presence of multiple extrema points in the function. In order to conduct a comparative study with other optimisation algorithms, the same test functions are solved using SOS, PSO and EP algorithms. The parameter of the optimisation algorithm used in this study is tabulated as in Table 1.

Table 2 lists the benchmark test function used in this paper. A 2-dimensional plot of the benchmark test functions are depicted as in Figure 2. For each test function, the optimisation algorithm will try to solve the benchmark test functions 20 times in order to observe the variation of results produced by the optimisation algorithm. The best fitness value yielded by the optimisation algorithms were tabulated as in Table 3.

Table 3 illustrates the best global minima value which is found by the optimisation algorithm. Functions labelled with U and M are unimodal and multimodal functions respectively. Separable functions are labelled with S, and non-separable functions are labelled with N. From the results obtained in Table 3, it can be observed that the proposed algorithm has managed to find the actual global minima point for most benchmark test function. However, the algorithm has failed to determine the actual global minima point for Dixon & Price function, Levy function and Rosenbrock function. Despite that, the proposed algorithm has yielded satisfactory results on the obtain global minima point of those functions.

Table 1: Parameters of the optimisation algorithm used in this paper

Algorithm	Parameter	Value
CISOS	Number of initial organisms	20
	Number of clones	10
	CLS mapping	Piecewise Linear Chaotic Map (PLCM)
	CLS control parameter	0.4
	Maximum CLS iteration	100
	Maximum CISOS iteration	1000
SOS	Number of organisms	20
	Maximum SOS iteration	1000
PSO	Number of particles	20
	Inertia weight	1.0
	Minimum particle velocity	-10
	Maximum particle velocity	10
	First acceleration coefficient	2
	Second acceleration coefficient	2
	Maximum PSO iteration	1000
EP	Number of individuals	20
	Search step	0.0015
	Maximum EP iteration	1000

Table 2: Benchmark test functions used in this paper

Function name	Expression	Search space	Global minima point
F1: Ackley (M,N)	$f_1(x) = -20\exp\left(-0.2\sqrt{\frac{1}{d}\sum_{i=1}^{d} x_i^2}\right) - \exp\left(\frac{1}{d}\sum_{i=1}^{d}\text{co}\right.$ exp(1)	[-32,32]	$f_1(x^*) = 0$ at $x^* = (0,\ldots,0)$
F2: Bohachevsky (M,S)	$f_2(x) = \sum_{i=1}^{d-1}\left[x_i^2 + 2x_{i+1}^2 - 0.3\cos(3\pi x_i) - 0.4\cos(4\pi\right.$	[-15,15]	$f_2(x^*) = 0$ at $x^* = (0,\ldots,0)$
F3: Dixon & Price (U,N)	$f_3(x) = (x_1 - 1)^2 + \sum_{i=2}^{d} i\left(2x_i^2 - x_{i-1}\right)^2$	[-10,10]	$f_3(x^*) = 0$ at $x_i^* = 2^{-\frac{2^i-2}{2^i}}$
F4: Griewank (M,N)	$f_4(x) = \sum_{i=1}^{d}\frac{x_i^2}{4000} - \prod_{i=1}^{d}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	[-600,600]	$f_4(x^*) = 0$ at $x^* = (0,\ldots,0)$
F5: Levy (M,N)	$f_5(x) = \sin^2(\pi w_1) + \sum_{i=1}^{d-1}\left[(w_i - 1)^2\left[1 + 10\sin^2(\pi w_i +\right.\right.$ where $a = (w_d - 1)^2\left[1 + \sin^2(2\pi w_d)\right]$, $w_i = 1 + \frac{x_i - 1}{4}$	[-10,10]	$f_5(x^*) = 0$ at $x^* = (1,\ldots,1)$
F6: Rastrigin (M,S)	$f_6(x) = 10d + \sum_{i=1}^{d}\left[x_i^2 - 10\cos(2\pi x_i)\right]$	[-5.12,5.12]	$f_6(x^*) = 0$ at $x^* = (0,\ldots,0)$
F7: Rosenbrock (U,N)	$f_7(x) = \sum_{i=1}^{d-1}\left[100\left(x_{i+1} - x_i^2\right)^2 + (x_i - 1)^2\right]$	[2.048,2.048]	$f_7(x^*) = 0$ at $x^* = (1,\ldots,1)$
F8: Schwefel No. 2.22 (U,S)	$f_8(x) = \sum_{i=1}^{d}\lvert x_i\rvert + \prod_{i=1}^{d}\lvert x_i\rvert$	[-10,10]	$f_8(x^*) = 0$ at $x^* = (0,\ldots,0)$
F9: Sphere (M,S)	$f_9(x) = \sum_{i=1}^{d} x_i^2$	[-5.12,5.12]	$f_9(x^*) = 0$ at $x^* = (0,\ldots,0)$
F10: Zakharov (M,N)	$f_{10}(x) = \sum_{i=1}^{d} x_i^2 + \left(\frac{1}{2}\sum_{i=1}^{d} ix_i\right)^2 + \left(\frac{1}{2}\sum_{i=1}^{d} ix_i\right)^4$	[-5,10]	$f_{10}(x^*) = 0$ at $x^* = (0,\ldots,0)$

From Table 3, it can be observed that PSO and EP have failed to find the actual global minima point of all benchmark test functions used in this study. By comparing the performance of PSO and EP in terms of finding the global minima point of the test functions, it is clear that PSO has performed significantly better compared to EP since the global minima point found by PSO are closer to the actual global minima point of the benchmark test functions as stated in Table 2. On the other hand, SOS algorithm has performed significantly better compared to PSO and EP through its successful attempt to find the global minima point of most benchmark test functions. However, it can be noted that CISOS algorithm has managed to find the global minima point of Zakharov function which SOS algorithm has failed to find the actual global minima point. It is also worth noted that CISOS also yielded lower global minima point value on Dixon & Price function, Levy function and Rosenbrock function.

Table 3: Results of best global minima value obtained by the optimisation algorithm

Test function	CISOS	SOS	PSO	EP
F1	**0**	**0**	13.22775	20.49759
F2	**0**	**0**	2.29358×10^{3}	7.53899×10^{3}
F3	$\mathbf{2.24978\times10^{-1}}$	6.66678×10^{-1}	1.88946×10^{6}	3.93933×10^{6}
F4	**0**	**0**	3.13898	9.23255×10^{3}
F5	$\mathbf{2.58859\times10^{-1}}$	5.77395×10^{-1}	2.00419×10^{2}	3.13963×10^{2}
F6	**0**	**0**	7.08392×10^{2}	7.02224×10^{2}
F7	**1.82859**	46.46912	1.00283×10^{4}	1.16655×10^{4}
F8	**0**	**0**	1.70231×10^{2}	7.45796×10^{18}
F9	**0**	**0**	2.42007×10^{2}	2.70986×10^{2}
F10	**0**	2.22070×10^{-233}	1.42523×10^{3}	1.54056×10^{10}

From the results, it can be seen that the functions which CISOS has failed to find its actual global minima point are non-separable functions. Non-separable functions are quite hard to be solved and pose a challenge to the optimisation algorithm to solve the functions. Multimodal functions can have a lot of local minima points, which poses a challenge to an optimisation algorithm to escape from the local minima. However, the proposed algorithm has managed to solve most multimodal test function, which indicates its capability to escape from local optima solution while solving optimisation problems. Therefore, it can be concluded that the proposed algorithm is more superior compared to other algorithms used for comparative study in this paper.

4. Conclusion

This paper presents an optimisation algorithm known as Chaotic Immune Symbiotic Organisms Search (CISOS) for solving optimisation problems. The hybridisation of CLS, AIS and SOS inspires the development of the proposed algorithm in the attempt to improve the searching capability of SOS. The performance of the proposed algorithm has been tested on 10 different benchmark test functions by assessing the global minima point provided by the algorithms with the actual global minima point of the function. CISOS has provided a high-quality solution, which reflects a superior performance compared to the other algorithms used for comparative study purposed in this paper. The proposed algorithm can be implemented to solve other optimisation problems which in turn can help to solve the problems with high-quality results.

Acknowledgement

The authors would like to acknowledge the Institute of Research Management and Innovation (IRMI), UiTM Shah Alam, Selangor, Malaysia for the financial support of this research. This research is supported by IRMI under BESTARI Research Grant Scheme with project code 600-IRMI/MYRA 5/3/BESTARI (027/2017).

References

[1] J. Li, D. Wang, W. Wang & J. Jiang, "Minimize Current Stress of Dual-Active-Bridge DC-DC Converters for Electric Vehicles Based on Lagrange Multipliers Method", *Energy Procedia*, Vol. 105, (2017), pp. 2733 – 2738.

[2] V. Torczon, R.M. Lewis & M.W. Trosset, "Direct search methods: then and now", *Journal of Computational and Applied Mathematics*, Vol. 124, No. 1-2, (2000), pp. 191 – 207.

[3] C. Hamzaçebi & F. Kutay, "Continuous functions minimization by dynamic random search technique", *Applied Mathematical Modelling*, Vol. 31, No. 10, (2007), pp. 2189 – 2198.

[4] J.R. Ribeiro Cavalcante & F.M. Campello de Souza, "A Global Optimization Algorithm for Solving Linear Programming Problem", *IFAC Proceedings Volumes*, Vol. 30, No. 19, (1997), pp. 513 – 515.

[5] L.L. Lai & J.T. Ma, "Application of evolutionary programming to reactive power planning-comparison with nonlinear programming approach", *IEEE Transactions on Power Systems*, Vol. 12, No. 1, (1997), pp. 198 – 206.

[6] K.Y. Lee & F.F. Yang, "Optimal reactive power planning using evolutionary algorithms: a comparative study for evolutionary programming, evolutionary strategy, genetic algorithm, and linear programming", *IEEE Transactions on Power Systems*, Vol. 13, No. 1, (1998), pp. 101 – 108.

[7] C.C. Asir Rajan, M. Surya Kalavathi & S. Ranganathan, "Self-adaptive firefly algorithm based multi-objectives for multi-type FACTS placement", *IET Generation, Transmission & Distribution*, Vol. 10, No. 11, (2016), pp. 2576 – 2584.

[8] T. Sen & H.D. Mathur, "A new approach to solve Economic Dispatch problem using a Hybrid ACO–ABC–HS optimization algorithm", *International Journal of Electrical Power & Energy Systems*, Vol. 78, (2016), pp. 735 – 744.

[9] M.T. Bhoskar, M.O.K. Kulkarni, M.N.K. Kulkarni, M.S.L. Patekar, G.M. Kakandikar & V.M. Nandedkar, "Genetic Algorithm and its Applications to Mechanical Engineering: A Review", *Materials Today: Proceedings*, Vol. 2, No. 4-5, (2015), pp. 2624 – 2630.

[10] C.M. Chan, H.L. Bai & D.Q. He, "Blade shape optimization of the Savonius wind turbine using a genetic algorithm", *Applied Energy*, Vol. 213, (2018), pp. 148 – 157.

[11] S. Parinam, A.L. Sharma, V.S.R.S. Praveen Kumar, M. Kumar, N. Kumari, S.K. Mittal & V. Karar, "Optimization of optical parameters for the design of multilayer bandpass filter using genetic algorithm", *Materials Today: Proceedings*, Vol. 5, No. 2, (2018), pp. 5091 – 5096.

[12] K.F. Fong, V.I. Hanby & T.T. Chow, "HVAC system optimization for energy management by evolutionary programming", *Energy and Buildings*, Vol. 38, No. 3, (2006), pp. 220 – 231.

[13] B.N.S. Rahimullah, E.I. Ramlan & T.K. Abdul Rahman, "Evolutionary approach for solving economic dispatch in power system", *Proceedings. National Power Engineering Conference, 2003. PECon 2003*, (2003), pp: 32 – 36.

[14] S. Domínguez-Isidro, E. Mezura-Montes & L.G. Osorio-Hernández, "Evolutionary programming for the length minimization of addition chains", *Engineering Applications of Artificial Intelligence*, Vol. 37, (2015), pp. 125 – 134.

[15] C.L. Liao, S.J. Lee, Y.S. Chiou, C.R. Lee, C.H. Lee, "Power consumption minimization by distributive particle swarm optimization for luminance control and its parallel implementations" *Expert Systems with Applications*, Vol. 96, (2018), pp. 479 – 491.

[16] A. Mortazavi & V. Toğan, "Sizing and layout design of truss structures under dynamic and static constraints with an integrated particle swarm optimization algorithm", *Applied Soft Computing*, Vol. 51, (2017), pp. 239 – 252.

[17] M. Awais, A. Basit, R. Adnan, Z.A. Khan, U. Qasim, T. Shafique & N. Javaid, "Overload management in transmission system using particle swarm optimization", *Procedia Computer Science*, Vol. 52, No. 1, (2015), pp. 858 – 865.

[18] T.K. Nizami & K. Sundareshwaran, "A feedback control design of buck converter: An artificial immune system based approach", *2015 39th National Systems Conference (NSC)*, (2015), pp: 1 – 6.

[19] K. Bamdad, M.E. Cholette, L. Guan & J. Bell, "Ant colony algorithm for building energy optimisation problems and comparison with benchmark algorithms", *Energy and Buildings*, Vol. 154, (2017), pp. 404 – 414.

[20] K. Rayudu, G. Yesuratnam & A. Jayalaxmi, "Artificial Bee Colony algorithm for optimal reactive power dispatch to improve voltage stability", *2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT)*, (2016), pp: 1 – 7.

[21] T. Apostolopoulos & A. Vlachos, "Application of the Firefly Algorithm for Solving the Economic Emissions Load Dispatch Problem", *International Journal of Combinatorics*, Vol. 2011, (2011), pp. 1 – 23.

[22] S. Yazdani, H. Nezamabadi-Pour & S. Kamyab, "A gravitational search algorithm for multimodal optimization", *Swarm and Evolutionary Computation*, Vol. 14, (2014), pp. 1 – 14.

[23] O. Ivanov O & M. Gavrilas, "A hybrid GA-PSO algorithm for static VAR compensation", *2016 International Conference and Exposition on Electrical and Power Engineering (EPE)*, (2016), pp: 681 – 686.

[24] B. Mahdad, K. Srairi & T. Bouktir, "Optimal power flow for large-scale power system with shunt FACTS using efficient parallel GA", *International Journal of Electrical Power & Energy Systems*, Vol. 32, No. 5, (2010), pp. 507 – 517.

[25] A. Soeprijanto & M. Abdillah, "Type 2 fuzzy adaptive binary particle swarm optimization for optimal placement and sizing of distributed generation", *2011 2nd International Conference on Instrumentation, Communications, Information Technology, and Biomedical Engineering*, (2011), pp: 233 – 238.

[26] R. Sirjani & A. Mohamed, "Improved Harmony Search Algorithm for Optimal Placement and Sizing of Static Var Compensators in Power Systems", *2011 First International Conference on Informatics and Computational Intelligence*, (2011), pp: 295 – 300.

[27] M. Cheng & D. Prayogo, "Symbiotic Organisms Search: A new metaheuristic optimization algorithm", *Computers & Structures*, Vol. 139, (2014), pp. 98 – 112.

[28] M.K. Mohamad Zamani, I. Musirin & S.I. Suliman, "Symbiotic Organisms Search Technique for SVC Installation in Voltage Control", *Indonesian Journal of Electrical Engineering and Computer Science*, Vol. 6, No. 2, (2017), pp. 318 – 329.

[29] D. Prasad & V. Mukherjee, "A novel symbiotic organisms search algorithm for optimal power flow of power system with FACTS devices", *Engineering Science and Technology, an International Journal*, Vol. 19, No. 1, (2016), pp. 79 – 89.

[30] U. Guvenc, S. Duman, M.K. Dosoglu, H.T. Kahraman, Y. Sonmez & C. Yilmaz, "Application of Symbiotic Organisms Search Algorithm to solve various economic load dispatch problems", *2016 International Symposium on INnovations in Intelligent SysTems and Applications (INISTA)*, (2016), pp: 1 – 7.

[31] S. Saha & V. Mukherjee, "Optimal placement and sizing of DGs in RDS using chaos embedded SOS algorithm", *IET Generation, Transmission & Distribution*, Vol. 10, No. 14, (2016), pp. 3671 – 3680.