A Chaotic Crow Search Algorithm for High-Dimensional Optimization Problems

Dunia S. Tahir
**Electrical Engineering Department**
**University of Basrah**
**Basrah, Iraq**
**Duniasattar680@gmail.com**

Ramzy S. Ali
**Electrical Engineering Department**
**University of Basrah**
**Basrah, Iraq**
**rsawaily@gmail.com**

***Abstract*- Crow Search Algorithm is an innovative meta-heuristic optimization algorithm. In this paper, chaotic maps are combined into Crow Search Algorithm to increase its global optimization. Ten variant chaotic maps are used and the Tent map is found as the best choices for high dimensional problems. The novel Chaotic Crow Search Algorithm is relied on the substitution of a random location of search space and the awareness parameter of crow with chaotic sequences. The results show that the chaotic maps are able to enhance the performance of the Crow Search Algorithm. Also the novel Chaotic Crow Search Algorithm outperforms the conventional Crow Search Algorithm, first version of Chaotic Crow Search Algorithm, Genetic Algorithm, and Particle Swarm Optimization Algorithm from the point view of speed convergence and the function dimensions.**

Index Terms— Chaos; Crow Search Algorithm; Chaotic Crow Search Algorithm; Meta-heuristic algorithms.

I. Introduction

In recent years, several novel meta-heuristic techniques have been proposed. These techniques are promised for solving difficult real-world optimization problems, like engineering design problems [1, 2, 3].The No Free Lunch theorem NFL proved that there is no algorithm for solving all optimization algorithms [4].This theorem is the groundwork of many studies in this field. Some of the best meta-heuristic algorithms are Genetic Algorithm (GA)[5],Particle Swarm Optimization (PSO) [6], Elephant herding optimization [7] , Monarch butterfly optimization [8], Kidney-inspired algorithm[9],Bat Algorithm (BA)[10] , Firefly Algorithm(FA) [11] , Krill Herd (KH) [12] and Crow Search Algorithm(CSA)[13] . Nonlinear dynamic techniques, exclusively chaos, can be attracted more consideration in many fields [14]. The applications of chaos in optimization algorithms represent one of the important fields [15]. The chaotic optimization algorithm is a special type of random optimization algorithms [16]. Chaotic optimization algorithm employs chaotic variables rather than random variables. Chaos has three important dynamic properties :(1) Ergodicity, (2)Regularity, and(3) Semi-stochastic property[17]. Chaotic maps are able to effectively improve the performance of meta-heuristic algorithms by fine-tuning parameters of these algorithms, as chaos-genetic algorithm [18], Chaos enhanced accelerated particle swarm algorithm[19], Chaotic harmony search algorithms [20], Chaotic cuckoo algorithm [21], Chaotic Gravitational Constants for GSA [22], Chaotic Imperialist competitive algorithm [23], Chaotic Firefly algorithm[16], and Chaotic bat swarm optimization [24]. Gehad I. and *et al*. are introduced the first version of Chaotic Crow Search algorithm (CCSA1)[25].The first version of chaotic crow search algorithm is suffered from low convergence rate with

high dimensional optimization problems.In this paper, second version of Chaotic Crow Search Algorithm (CCSA2) is proposed to improve the performance of both the conventional crow search algorithm (CSA)[13] and the first version of chaotic crow search algorithm (CCSA1)[25]. The main idea of the proposed algorithm is to use chaotic maps to initialize the awareness probability and the location of search space of crow instead of randomly initialization. In order to estimate the proposed algorithm, three unimodal and four multimodal benchmark functions are used to confirm the proposed algorithm. The paper is prepared as follows: Section II deals with the crow search algorithm, Section III defines chaos and some well-known chaotic maps, and the proposed algorithm is explained in Section IV, Section V describes the statistical calculations, results and discussion in Section VI, and Section VII have represented the conclusion.

II. Crow Search algorithm

Crow search algorithm (CSA) which was first introduced by A. Askarzadeh(2016), is inspired by the cleverness of crows for storing their additional nutrition in special storage locations and regain it when the food is required[13].The CSA algorithm is working according to the following principles:

The number of crows is referred to as flock size (N) and the location of crow (β) at time iteration(itera) in the search space is identified by a vector $x^{\beta,itera}$ (β=1,2 ,… ..,N); itera=(1,2..,$itera_{max}$) where $x^{\beta,itera}=[x_1^{\beta,itera}, x_2^{\beta,itera} \ldots, x_d^{\beta,itera}]$, (d) is the number of the decision variables and $itera_{max}$ is referred to the maximum number of generations . The crows are saved the best locations of concealment food in their memory.

At iteration (itera), the storage location of the food of crow β is presented by ($m^{\beta,itera}$),where $m^{\beta,itera}$ is the memory of crow β at iteration itera. This is the best storage location that crow β has found until now. The resources of better food location are searched by crows that are walked in the environment. Assume that at iteration (itera), crow (α) desires to visit its storing location, ($m^{\alpha,itera}$) is the memory of crow α at iteration (itera) . At this iteration, crow (β) decides to follow crow α to discover the storage location of crow α. So, two modes may occur:

Mode one: If Crow α is unaware of being followed by crow β ,crow β will modify its location of crow α as follows [13]:

$$x^{\beta,itera+1} = x^{\beta,itera} + r_\beta * fl^{\beta,itera} * (m^{\alpha,itera} - x^{\beta,itera}) \quad (1)$$

Where r_β is a random number implemented by the standard uniform distribution and $fl^{\beta,itera}$ indicates the flying length of crow β at iteration(itera).

Mode two: If Crow α is aware of being followed by crow β ,it will fool Crow β by randomly going to aposition of the search space. Completely, modes one and two can be calculated as follows [13]:

$$x^{\beta,itera+1} = \begin{cases} x^{\beta,itera} + r_\beta * fl^{\beta,itera} * (m^{\alpha,itera} - x^{\beta,itera}) & r_\alpha \geq AP \\ a \text{ random location} & r_\alpha < AP \end{cases} \quad (2)$$

Where r_α is a random number implemented by the standard uniform distribution and $AP^{\alpha,itera}$ indicates the awareness probability of crow α at iteration (itera).

Crow search algorithm

Randomly initialize the location of a flock of N crows in the search space

Evaluate the location of the crows

Initialize the memory of each crow

while $itera < itera_{max}$

for $\beta = 1 : N$ (*all N crows of the flock*)

Randomly choose one of the crows to follow (*for example* α)

Define an awareness probability

if $r_\alpha \geq AP^{\alpha,itera}$

$x^{\beta,itera+1} = x^{\beta,itera} + r_\beta * fl^{\beta,itera} * (m^{\alpha,itera} - x^{\beta,itera})$

else

$x^{\beta,itera+1}$ = *a random location of search area*

end of if

end of for

The feasibility of new locations is checked

Evaluate the new location of the crows

Update the memory of crows

end of while

Fig. 1. Procedure of the CSA algorithm

III. CHAOTIC MAPS

To avoid local optima and accomplish better results, a well dispersion which is the important characteristics of chaotic maps is a good solution to improve the performance of meta-heuristic algorithms [26].

In this paper, one dimension, non-invertible maps are used to produce chaotic maps. Ten variant 1-D maps [27] are described in Table I. In this table, j means the index of the chaotic sequence, and x_j represents the jth number in the chaotic sequence. The initial value 0.7 was chosen for all maps that are used in this study.

IV. CHAOTIC CROW SEARCH ALGORITHM

Intensification and diversification are two major components of any meta-heuristic algorithms [4]. Intensification (local search) examines about the existent preferable solutions and takes the preferable one, while diversification (global search) permits for the meta-heuristic algorithms to investigate more efficiently in the search space. All meta-heuristic algorithms have a facility to balance between intensification and diversification is very well [12]. In crow search algorithm (CSA), exploitation and

TABLE I TEN VARIANT CHAOTIC MAPS

No.	Name	Function
1	Chebyshev map	$x_{j+1} = \cos(j \cos^{-1}(x_j))$
2	Circle map	$x_{j+1} = \text{mod}(x_j + b - (a/(2\pi x_j), 1))$ a=0.5 and b=0.2
3	Gauss map	$x_{j+1} = \begin{cases} 0 & x_j = 0 \\ 1/(\text{mod}(x_j, 1)) & \text{otherwise} \end{cases}$
4	Iterative map	$x_{j+1} = \sin(a\pi / x_j)$, a=0.7
5	Logistic map	$x_{j+1} = a\, x_j(1 - x_j)$, a=4
6	Piecewise map	$x_{j+1} = \begin{cases} x_j/a & 0 \leq x_j < a \\ (x_j - a)/(0.5-a) & a \leq x_j < 0.5 \\ (1-a-x_j)/(0.5-a) & 0.5 \leq x_j < 1-a \\ (1-x_j)/a & 1-a \leq x_j < 1 \end{cases}$ a=0.4
7	Sine map	$x_{j+1} = (a/4)\sin(\pi x_j)$, a=4
8	Singer map	$x_{j+1} = \mu(7.86 x_j - 23.31 x_j^2 + 28.75 x_j^3 - 13.302875 x_j^4)$, μ=1.07
9	Sinusoidal map	$x_{j+1} = a\, x_j^2 \sin(\pi x_j)$, a=2.3
10	Tent map	$x_{j+1} = \begin{cases} x_j/0.7 & x_j < 0.7 \\ (10/3)(1-x_j) & x_j \geq 0.7 \end{cases}$

exploration are essentially dominated by the parameter of awareness probability (AP) Therefore, using small values of AP, increases intensification and vice versa. Generally, one of the methods of utilizing chaos in meta-heuristic algorithms is to use them instead of random values and probabilities. In the other words, chaotic maps are switched by random values and probabilities to provide chaotic behaviors for meta-heuristic algorithms.

In conventional CSA algorithm, crows transport from place to another and look for better storage locations of sustenance. The crows are stored the best knowledge about the location of food in their memory. A. Askarzadeh mentions two modes for finding the best location of concealment food according to the awareness probability of crow [13].

In this paper, the random location and awareness probability values are substituted by chaotic maps as follows:

Chaotic Crow search algorithm (CCSA2)

Randomly initialize the location of a flock of N crows in the search space

Evaluate the location of the crows

Initialize the memory of each crow

while $itera < itera_{max}$

for $\beta = 1 : N$ (*all N crows of the flock*)

Randomly choose one of the crows to follow (*for example* α)

Define an awareness probability

if $r_\alpha \geq CM^{\alpha,itera}$

$x^{\beta,itera+1} = x^{\beta,itera} + r_\beta * fl^{\beta,itera} * (m^{\alpha,itera} - x^{\beta,itera})$

else

$x^{\beta,itera+1} = CM^{\alpha,itera}$

end of if

end of for

The feasibility of new locations is checked
Evaluate the new location of the crows
Update the memory of crows
end of while

Fig. 2. Procedure of the CCSA2 algorithm.

Where CM is the chaotic map. In conventional CSA algorithm, AP denotes the awareness probability of crow which is chosen as 0.1 to find the best results. Visibly, employing the constant value is not more optimum for the problems. As a result, the chaotic maps are used to provide the chaotic manners for the crow search algorithm. So, in Chaotic Crow Search algorithm (CCSA2), awareness probability of crow is changed to chaotic sequences and the second state for finding the best location of crow in standard CSA algorithm is chosen as a random number while in chaotic crow search algorithm(CCSA2) is switched to chaotic maps. With chaotic maps, searches to find the best solution can perform rapidly compared to random searches.

V. CRITERION FOR PERFORMANCE MEASUREMENTS

Many criteria for evaluating the performance of the algorithms are found. In this paper, statistical measurements and success rate will calculate. The statistical measurements represent the best solution, mean and standard deviation. Success Rate (SR) represents another criterion for performance measure, is defined as:

$$SR=100*(X/Y) \quad (3)$$

Where X is the successful solution which is found and Y is the value of all executions. In this paper, an execution is considered as a successful execution when it is very near to the global optimum. Thus, the criteria for an effective execution can be characterized as [28]:

$$|Xgbest – Xopti| \le Z \quad (4)$$

Where Xgbest is the obtained global best result of the algorithm and *Xopti* is the global optimum. In this study, *Z* is equal to 10^{-4} or 10^{-5} which represents the tolerance of success rate.

VI. RESULTS AND DISCUSSION

Gehad I. and *et al*. are presented the first version of Chaotic Crow Search Algorithm (CCSA1) [25]. It deals with low dimensional optimization problems. In this paper high dimensional optimization problems are used to evaluate the efficiency of the Chaotic Crow Search Algorithm (CCSA2), seven standard benchmark functions are selected [29]. Three benchmark functions are unimodal which are used to examine intensification while the others all multimodal functions are suitable for benchmarking diversification. Table II lists these functions, where the range is the limit of the function's search space and D indicates the dimension of the functions. The global minimum values of all objectives are equal to zero.

In CSA, CCSA1, CCSA2, GA, and PSO algorithms, the population size is 20 and the number of iterations is 2000. The flight length of crow (fl) value for the CSA , CCSA1 and CCSA2 algorithms has been set to two while the awareness probability (AP) of crow value is set to 0.1 for both the CSA and CCSA1 algorithms, and using chaotic sequences for CCSA2 algorithm. In PSO, cognitive and social constant parameters are set to two while inertia weight decreases linearly from 0.9 to 0.4 during iterations. In GA, roulette wheel selection, crossover with the coefficients of 0.7 and 0.5 and uniform mutation are selected.

TABLE II BENCHMARK FUNCTIONS

FN#	Name	Range	D	Type
FN1	Alpine	[-10,10]	30,50	Multimodal
FN2	Rastrigin	[-5.12,5.12]	30,50	Multimodal
FN3	Salomon	[-100,100]	30,50	Multimodal
FN4	Schwelefl 2.21	[-100,100]	30,50	Unimodal
FN5	Schwelefl 2.22	[-10,10]	30,50	Unimodal
FN6	Sphere	[-100,100]	30,50	Unimodal
FN7	Wavy	[-π,π]	30,50	Multimodal

All results are completed by MATLAB on Core 2 Duo 2.67 GHz HP laptop. Tables III, IV,.., and VIII show the experimental results obtained by CCSA1 and CCSA2 algorithms after (30) independent runs at 30 and 50 dimensions. In these tables, the values of best, mean and standard deviation are measured to illustrate the performance of the chaotic crow search algorithm. Figures 3 to 16 show the convergence curves of the CSA, CCSA1, CCSA2, GA, and PSO algorithms. It is clear from figures that CCSA2 outperforms the CSA, CCSA1,GA, and PSO algorithms in two points. The first is the speed of convergence and the second is the function dimension.

In this paper, Z is equal to 10^{-4} or 10^{-5} which represents the tolerance of success rate. Tables IX and X show the success rate values of CCSA1 and CCSA2 algorithms for seven benchmark functions with ten variant chaotic maps for 30 and 50 dimensions. When the results of success rates in Tables IX and X examined, it is clear that the CCSA2 is outperformed the CCSA1 for all chaotic maps. The results of the Tables XI and XII showed a comparison of the best ,mean and standard deviation of CSA,CCSA1,CCSA2,GA ,and PSO algorithms. In all functions the convergence of the CCSA2 much faster than all other algorithms(CSA,CCSA1,GA and PSO) expected in function 5 (FN5) at dimension 30 the PSO algorithm converges faster than CCSA2. The results of the CCSA1and CCSA2 algorithms are applied to seven variant benchmark functions with ten chaotic maps that approve its significant outperformance over conventional CSA, GA, and PSO algorithms at higher dimensional and show that Tent map causes a good balance between intensification and diversification for CCSA2. While CCSA1 show a good balance between intensification and diversification in Logistic map.

VII. CONCLUSION

The paper presents a new meta-heuristic algorithm which is called the chaotic crow search algorithm (CCSA2). A set consists of seven test functions benchmarked the performance of the algorithm. Ten chaotic maps were used to define the awareness parameter and the location of search space of crow. The results proved that the proposed algorithm can develop the performance of both the conventional crow search algorithm(CSA) and the first version of chaotic crow search algorithm(CCSA1) within the proposed algorithm by avoiding local optimum, increasing the speed of reaching to the global solution, and solving problems with high dimensional. The results showed that the chaotic crow search algorithm (CCSA2) has a good balancing between intensifcation and diversification and it is outperformed the conventional CSA,CCSA1, GA, PSO algorithms. Moreover, the results show that the CCSA1 with Logistic map causes a good balance between intensification and diversification while in the second version of chaotic crow search algorithm (CCSA2) the Tent map achieved a good balance. For future research, solving real-world electrical and computer engineering problems will use to investigate the performance of the CCSA2 algorithm.

Table III. Best Values With Ten Variant Chaotic Maps For 30d

FN#	Methods	Chaotic Map No.									
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
FN1	CCSA1	1.3966e-01	1.4894e-02	1.6526e+01	2.8518e-02	**1.2333e-02**	4.1724e-02	3.2246e-02	6.8295e-03	2.1050e+01	3.4527e-02
	CCSA2	8.1644e-07	9.0039e-09	4.1776e-07	**5.5726e-08**	4.4652e-07	1.0258e-07	7.3681e-06	6.3794e-06	7.1288e-08	2.1274e-07
FN2	CCSA1	1.6997e+01	1.5919e+01	1.6594e+02	1.6920e+01	1.4962e+01	1.2936e+01	1.4974e+01	1.4927e+01	1.5660e+02	**1.0080e+01**
	CCSA2	3.4106e-12	**5.5422e-12**	7.3008e-12	2.5899e-11	**1.5987e-13**	1.9184e-12	8.5265e-13	7.8776e-09	3.9637e-10	1.5027e-11
FN3	CCSA1	5.9987e-01	6.9987e-01	6.9223e+00	**5.9987e-01**	5.9988e-01	6.9987e-01	5.9987e-01	8.9987e-01	8.2998e+00	1.2998e+00
	CCSA2	5.3664e-06	3.4017e-08	3.1266e-08	3.2896e-07	7.5885e-07	2.4288e-07	**8.8469e-09**	2.2972e-05	1.4859e-05	1.0439e-08
FN4	CCSA1	2.7260e+00	2.9746e+00	2.1008e+01	1.6933e+00	**9.2891e-01**	1.5087e+00	1.1252e+00	1.7447e+00	2.7566e+01	2.7997e+00
	CCSA2	1.0336e-07	8.0762e-07	1.2890e-06	1.4689e-07	1.3925e-06	1.0774e-06	**2.3161e-08**	2.9398e-05	4.3711e-05	9.5472e-08
FN5	CCSA1	1.1458e+00	9.1912e-01	2.8766e+01	4.6405e-01	**3.1899e-01**	6.2554e-01	5.0281e-01	3.6941e-01	3.5452e+01	9.4470e-01
	CCSA2	2.6021e-05	1.4566e-05	5.3795e-05	**4.9710e-08**	3.3927e-06	1.8889e-06	1.4763e-05	1.0778e-04	4.8865e-05	4.6408e-06
FN6	CCSA1	5.3552e-02	**5.3202e-06**	3.8585e+03	4.4710e-03	2.7175e-02	2.0299e-03	2.8436e-02	1.4372e-03	4.7351e+03	9.6365e-03
	CCSA2	2.2729e-10	9.0297e-10	9.7049e-11	1.5232e-13	1.5280e-12	1.1889e-10	4.1146e-11	7.8484e-11	7.2893e-11	**5.8709e-12**
FN7	CCSA1	**9.7187e-02**	1.0092e-01	5.0992e-01	1.7063e-01	1.5918e-01	1.1908e-01	1.2747e-01	1.7949e-01	6.3435e-01	1.3720e-01
	CCSA2	3.4416e-15	**1.1102e-16**	2.8865e-15	1.1102e-16	1.5187e-13	3.5527e-15	2.2204e-16	1.8762e-14	2.3658e-12	2.2204e-16

Table IV. Mean Values With Ten Variant Chaotic Maps For 30d

FN#	Methods	Chaotic Map No.									
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
FN1	CCSA1	1.9845e+00	1.5593e+00	1.8351e+01	1.2685e+00	8.5366e-01	1.5868e+00	9.1398e-01	1.6444e+00	4.2865e+01	2.0091e+00
	CCSA2	1.0804e-04	2.6786e-05	8.4970e-06	2.8665e-06	2.5159e-05	1.6961e-05	1.8480e-05	1.5278e-04	5.8064e-05	1.1819e-05
FN2	CCSA1	3.6429e+01	3.0320e+01	1.9224e+02	4.1367e+01	3.1257e+01	2.6310e+01	3.3354e+01	3.1810e+01	2.8434e+02	2.6797e+01
	CCSA2	1.5764e-08	4.1393e-07	6.3507e-08	1.4966e-06	2.9201e-07	1.8492e-07	1.6477e-07	1.1623e-05	2.9062e-06	1.0064e-07
FN3	CCSA1	9.3928e-01	1.0302e+00	7.7435e+00	7.6367e-01	8.2865e-01	9.5987e-01	8.5298e-01	1.1566e+00	1.2883e+01	1.6352e+00
	CCSA2	1.3425e-02	7.3404e-05	3.5239e-03	2.2298e-05	1.7157e-04	1.4009e-05	3.3961e-03	2.6101e-03	7.0097e-04	1.0186e-04
FN4	CCSA1	5.0325e+00	5.6951e+00	2.5668e+01	4.3504e+00	2.6840e+00	5.7812e+00	3.0720e+00	3.0861e+00	4.0220e+01	5.5524e+00
	CCSA2	3.3364e-05	1.1411e-04	2.4676e-04	3.4901e-04	4.7935e-05	1.9719e-05	2.6786e-05	1.2056e-03	2.8522e-04	3.2013e-05
FN5	CCSA1	2.6119e+00	2.3657e+00	3.1659e+01	1.7912e+00	1.2535e+00	2.4993e+00	1.5703e+00	1.7170e+00	5.0274e+01	2.1786e+00
	CCSA2	6.3179e-04	4.7513e-04	2.9110e-03	9.6788e-05	5.4836e-04	5.9632e-04	2.1068e-03	3.0426e-03	9.1833e-04	2.2539e-04
FN6	CCSA1	1.9773e-01	1.9262e-05	5.3492e+03	1.3758e-02	9.1742e-02	1.1884e-02	9.0989e-02	1.3742e-02	1.4793e+04	7.5269e-02
	CCSA2	5.6535e-04	7.5700e-05	2.1748e-06	0.0037	8.6580e-08	1.1472e-05	5.6280e-07	6.1394e-05	8.6238e-06	4.1725e-06
FN7	CCSA1	3.3588e-01	2.4974e-01	6.0724e-01	3.6695e-01	2.7402e-01	2.7588e-01	2.9715e-01	3.5173e-01	7.4684e-01	2.4812e-01
	CCSA2	3.7351e-11	2.2149e-10	4.2961e-10	5.0036e-11	1.3014e-09	8.4548e-10	7.6927e-11	4.7109e-08	4.2027e-09	5.7458e-11

Table V. Standard Deviation Values With Ten Variant Chaotic Maps For 30d

FN#	Methods	Chaotic Map No.									
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
FN1	CCSA1	1.9432e+00	1.6196e+00	1.0320e+00	1.6806e+00	9.5603e-01	1.4004e+00	7.7188e-01	1.6860e+00	9.7849e+00	2.1436e+00
	CCSA2	1.6576e-04	2.6881e-05	1.0401e-05	4.1129e-06	2.3893e-05	1.9486e-05	1.1000e-05	1.3329e-04	6.5144e-05	2.6827e-05
FN2	CCSA1	1.1398e+01	1.4939e+01	1.1755e+01	1.4773e+01	1.1459e+01	7.4570e+00	9.8451e+00	1.3708e+01	6.9599e+01	9.5278e+00
	CCSA2	4.9049e-08	1.0753e-06	1.9496e-07	4.4041e-06	5.8813e-07	8.7966e-07	5.9275e-07	1.6895e-05	6.4771e-06	3.1825e-07
FN3	CCSA1	1.4787e-01	1.6932e-01	3.8738e-01	9.1210e-02	1.0847e-01	1.6451e-01	1.0648e-01	1.9088e-01	2.4706e+00	2.1242e-01
	CCSA2	2.9808e-02	1.9093e-04	1.7899e-02	4.9745e-05	4.8228e-04	1.5271e-05	1.7917e-02	2.9901e-03	7.5165e-04	3.9811e-04
FN4	CCSA1	1.3707e+00	1.4217e+00	1.6077e+00	1.6511e+00	1.1454e+00	1.6030e+00	1.0533e+00	1.0560e+00	8.2292e+00	1.4427e+00
	CCSA2	4.4178e-05	2.1742e-04	7.7218e-04	4.7219e-04	6.2792e-05	2.6649e-05	5.5441e-05	1.3298e-03	2.6034e-04	1.0146e-04
FN5	CCSA1	1.0260e+00	8.1031e-01	1.5484e+00	7.8005e-01	6.1896e-01	1.4100e+00	7.2037e-01	8.7510e-01	1.0128e+01	6.7586e-01
	CCSA2	8.5928e-04	5.5415e-04	3.4459e-03	1.1192e-04	8.0063e-04	7.8149e-04	4.4146e-03	2.4509e-03	9.7383e-04	2.5045e-04
FN6	CCSA1	8.5449e-02	9.8624e-06	4.7602e+02	7.5189e-03	4.2211e-02	8.6963e-03	3.5273e-02	9.0379e-03	6.3665e+03	6.5460e-02
	CCSA2	0.0020	2.8759e-04	5.8243e-06	0.0199	2.3961e-07	4.7540e-05	1.2566e-06	1.0978e-04	1.4834e-05	4.1725e-06
FN7	CCSA1	1.0753e-01	8.3179e-02	3.5083e-02	1.1430e-01	6.9350e-02	1.0237e-01	7.6438e-02	8.7817e-02	5.2070e-02	7.3889e-02
	CCSA2	7.1656e-11	5.5167e-10	1.4684e-09	2.1398e-10	5.1429e-09	2.2622e-09	2.1409e-10	7.2937e-08	5.8429e-09	2.0172e-10

Table VI. Best Values With Ten Variant Chaotic Maps For 50d

FN#	Methods	Chaotic Map No.									
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
FN1	CCSA1	7.3195e-01	4.3149e-02	3.4772e+01	**4.0369e-02**	2.9852e-01	9.4085e-01	1.9480e-01	2.1729e-01	3.5338e+01	2.4266e-01
	CCSA2	1.1459e-06	6.3957e-07	1.1331e-07	1.3990e-07	3.4485e-07	5.3911e-07	3.9695e-06	8.2116e-06	1.2661e-01	**9.4178e-08**
FN2	CCSA1	3.7425e+01	3.0851e+01	3.3022e+02	2.8471e+01	2.6453e+01	**2.0206e+01**	2.6669e+01	3.4576e+01	3.9161e+02	2.2610e+01
	CCSA2	4.8761e-11	9.3258e-12	1.9895e-11	1.7763e-13	1.3056e-11	2.2471e-11	1.5187e-11	1.6049e-09	8.2689e-11	**0**
FN3	CCSA1	1.3998e+00	1.3998e+00	1.0324e+01	**1.1998e+00**	1.3305e+00	1.3998e+00	1.4046e+00	1.6567e+00	1.1699e+01	2.1998e+00
	CCSA2	1.8051e-06	1.2796e-07	**6.3946e-09**	5.1802e-07	1.9678e-07	1.1384e-06	6.1517e-08	5.5156e-07	4.7164e-06	1.0663e-08
FN4	CCSA1	6.3846e+00	7.5180e+00	3.1190e+01	5.1662e+00	**4.2684e+00**	7.1541e+00	4.5084e+00	4.4746e+00	2.9506e+01	5.4938e+00
	CCSA2	2.3827e-06	2.4886e-06	2.1522e-07	**5.5298e-08**	1.2765e-06	6.6724e-07	5.6991e-08	3.0333e-05	1.0409e-05	8.1223e-08
FN5	CCSA1	2.7056e+00	3.0838e+00	5.3229e+01	2.6518e+00	**2.3732e+00**	3.2151e+00	2.5812e+00	2.5660e+00	7.2096e+01	3.8826e+00
	CCSA2	1.0017e-05	2.6641e-05	**2.6513e-07**	1.7376e-06	5.4606e-05	5.2431e-06	3.6743e-06	2.6384e-06	4.0577e-05	5.7457e-06
FN6	CCSA1	2.1590e+00	**6.5224e-03**	1.0658e+04	2.9256e-01	1.2102e+00	9.0305e-01	2.7059e+00	4.8509e-01	1.7401e+04	2.3367e+00
	CCSA2	1.7647e-09	1.1278e-11	9.3503e-10	1.2650e-11	3.6082e-10	5.5641e-11	4.0375e-11	1.0074e-07	3.3193e-09	**9.6058e-12**
FN7	CCSA1	1.7448e-01	1.2806e-01	6.5859e-01	2.0931e-01	1.2266e-01	1.5939e-01	1.6191e-01	1.9974e-01	7.2406e-01	**1.2005e-01**
	CCSA2	7.6605e-15	5.0459e-13	0	1.1102e-16	1.3192e-12	4.7739e-15	1.1102e-16	5.0165e-12	1.2432e-11	**0**

Table VII. Mean Values With Ten Variant Chaotic Maps For 50d

FN#	Methods	Chaotic Map No.									
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
FN1	CCSA1	6.8665e+00	3.8071e+00	3.7507e+01	4.1700e+00	3.3861e+00	4.8485e+00	2.8492e+00	4.5758e+00	8.2400e+01	4.6610e+00
	CCSA2	2.7985e-05	3.8066e-05	1.9881e-04	6.5939e-06	4.0622e-05	9.4938e-05	9.5337e-05	2.9896e-04	1.1202	9.3449e-06
FN2	CCSA1	6.6985e+01	4.8843e+01	3.8037e+02	6.4180e+01	5.6052e+01	4.5936e+01	4.8476e+01	6.0114e+01	5.7775e+02	4.6361e+01
	CCSA2	2.7411e-07	2.1862e-07	1.7520e-05	8.3657e-08	4.7169e-07	6.3249e-05	4.7698e-07	2.9374e-05	1.4981e-06	3.2650e-08
FN3	CCSA1	1.7506e+00	1.8665e+00	1.1459e+01	1.5232e+00	1.5957e+00	1.8665e+00	1.7568e+00	2.0821e+00	1.8353e+01	2.7238e+00
	CCSA2	4.4995e-03	4.3896e-05	1.7673e-05	4.1131e-03	3.7769e-04	1.3655e-03	5.4099e-04	4.3542e-03	2.5392e-04	3.4267e-03
FN4	CCSA1	8.6890e+00	9.2425e+00	3.3019e+01	7.8361e+00	6.4949e+00	8.8071e+00	6.6827e+00	6.7718e+00	4.2734e+01	8.8864e+00
	CCSA2	1.0826e-03	1.5040e-04	6.1275e-05	1.2829e-05	8.3317e-05	1.5861e-05	2.0582e-05	9.3923e-04	3.1285e-04	2.4150e-05
FN5	CCSA1	5.5038e+00	4.7754e+00	6.1032e+01	4.4409e+00	4.3141e+00	5.2810e+00	4.3211e+00	3.9973e+00	9.9628e+01	5.9658e+00
	CCSA2	3.0611e-03	7.4591e-04	1.7501e-04	6.5159e-04	2.4904e-03	1.2674e-03	2.3529e-03	4.5152e-03	1.5770e-03	4.2711e-04
FN6	CCSA1	5.3496e+00	2.2341e-02	1.2059e+04	9.5212e-01	2.7379e+00	1.7127e+00	4.1116e+00	1.7942e+00	2.6883e+04	5.8109e+00
	CCSA2	1.2897e-04	1.0983e-05	4.6191e-04	3.4821e-08	3.9898e-05	1.5294e-06	1.4121e-05	1.2236e-04	8.7710e-06	4.3259e-07
FN7	CCSA1	3.5701e-01	2.5740e-01	6.9875e-01	4.0684e-01	2.8013e-01	2.5546e-01	3.0421e-01	3.5406e-01	8.2051e-01	2.2944e-01
	CCSA2	2.6581e-11	7.8922e-10	1.2464e-10	2.2304e-09	1.6647e-09	1.2757e-11	1.0044e-10	4.8006e-08	7.6679e-09	2.2040e-11

Table VIII. Standard Deviation Values With Ten Variant Chaotic Maps For 50d

FN#	Methods	Chaotic Map No.									
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
FN1	CCSA1	4.1905e+00	2.8757e+00	1.2722e+00	3.7788e+00	2.7821e+00	3.3937e+00	2.3224e+00	2.5394e+00	1.9749e+01	3.5154e+00
	CCSA2	3.0619e-05	9.3671e-05	4.0683e-04	6.0672e-06	5.2872e-05	1.3313e-04	1.1211e-04	2.6033e-04	4.3426e-01	9.9252e-06
FN2	CCSA1	1.7520e+01	1.3363e+01	1.7081e+01	1.8527e+01	1.8631e+01	1.3231e+01	1.1300e+01	1.8806e+01	1.0591e+02	1.4552e+01
	CCSA2	7.4044e-07	9.5843e-07	6.4066e-05	2.1666e-07	2.4698e-06	3.3188e-04	1.8151e-06	5.2553e-05	1.6404e-06	1.5328e-07
FN3	CCSA1	1.9810e-01	2.1187e-01	3.8712e-01	1.7258e-01	1.3702e-01	2.1029e-01	2.0511e-01	2.4134e-01	3.2371e+00	2.8809e-01
	CCSA2	1.8056e-02	6.5824e-05	2.3912e-05	1.7864e-02	1.3178e-03	4.5254e-03	2.5862e-03	1.4591e-02	2.9143e-04	1.7912e-02
FN4	CCSA1	1.2180e+00	1.1005e+00	1.0953e+00	1.3728e+00	1.2175e+00	9.7029e-01	1.1392e+00	1.1657e+00	6.3443e+00	1.2211e+00
	CCSA2	4.0451e-03	5.7255e-04	1.7668e-04	1.8905e-05	7.6337e-05	1.9721e-05	3.8004e-05	7.5351e-04	2.6396e-04	6.3141e-05
FN5	CCSA1	1.2669e+00	1.2554e+00	2.9805e+00	1.0095e+00	1.4229e+00	1.2258e+00	8.7052e-01	1.0986e+00	2.0651e+01	1.1594e+00
	CCSA2	5.5193e-03	6.2270e-04	2.0145e-04	1.0581e-03	3.6319e-03	2.2948e-03	4.8471e-03	3.8572e-03	2.2534e-03	6.2293e-04
FN6	CCSA1	1.5790e+00	1.0676e-02	7.2194e+02	3.8988e-01	9.5837e-01	7.5066e-01	1.3258e+00	8.9229e-01	8.7031e+03	1.8510e+00
	CCSA2	5.3022e-04	4.7788e-05	2.1011e-03	6.8045e-08	1.2308e-04	6.4683e-06	3.9629e-05	2.4038e-04	1.1751e-05	9.9383e-07
FN7	CCSA1	1.0143e-01	8.2083e-02	1.9452e-02	9.9706e-02	7.8369e-02	5.2551e-02	7.1948e-02	8.5499e-02	3.8730e-02	7.4348e-02
	CCSA2	5.2046e-11	1.9696e-09	2.5751e-10	6.7995e-09	4.2948e-09	2.6984e-11	4.9979e-10	7.9757e-08	8.6876e-09	7.3236e-11

Table IX . Success Rate Values Of Ccsa For Seven Benchmark Functions With Ten Variant Chaotic Maps For 30d

FN#	Methods	Success Rate with tolerance Z=10^{-4}										Success Rate with tolerance Z=10^{-5}									
		Chaotic Map No.										Chaotic Map No.									
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
FN1	CCSA1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	CCSA2	96	96	63	100	93	70	70	30	0	100	20	43	33	80	26	30	10	03	0	66
FN2	CCSA1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	CCSA2	100	96	93	100	100	96	100	90	100	100	100	90	90	100	96	90	100	43	100	100
FN3	CCSA1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	CCSA2	63	96	96	60	76	50	76	26	40	80	26	30	60	16	16	13	53	03	03	56
FN4	CCSA1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	CCSA2	36	86	90	100	63	100	93	10	23	93	16	50	50	66	06	53	60	0	0	66
FN5	CCSA1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	CCSA2	10	13	50	30	03	26	23	06	13	33	0	0	13	06	0	03	03	03	0	10
FN6	CCSA1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	CCSA2	86	96	90	100	93	100	96	73	100	100	66	93	76	100	76	96	76	50	63	100
FN7	CCSA1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	CCSA2	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100

Table X. Success Rate Values Of Ccsa For Seven Benchmark Functions With Ten Variant Chaotic Maps For 50d

FN#	Methods	Success Rate with tolerance Z=10^{-4}										Success Rate with tolerance Z=10^{-5}									
		Chaotic Map No.										Chaotic Map No.									
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
FN1	CCSA1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	CCSA2	96	96	63	100	93	70	70	30	0	100	20	43	33	80	26	30	10	03	0	66
FN2	CCSA1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	CCSA2	100	96	93	100	100	96	100	90	100	100	100	90	90	100	96	90	100	43	100	100
FN3	CCSA1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	CCSA2	63	96	96	60	76	50	76	26	40	80	26	30	60	16	16	13	53	03	03	56
FN4	CCSA1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	CCSA2	36	86	90	100	63	100	93	10	23	93	16	50	50	66	06	53	60	0	0	66
FN5	CCSA1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	CCSA2	10	13	50	30	03	26	23	06	13	33	0	0	13	06	0	03	03	03	0	10
FN6	CCSA1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	CCSA2	86	96	90	100	93	100	96	73	100	100	66	93	76	100	76	96	76	50	63	100
FN7	CCSA1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	CCSA2	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100

Table XI. Comparison Of Csa, Ccsa1, Ccsa2, Ga, And Pso For Variant Test Functions With 30d

FN#	Method	Best	Mean	Std.	SR
FN1	CSA	6.7385e-02	1.7204e+00	2.1001e+00	0
	CCSA1	1.2333e-02	8.5366e-01	9.5603e-01	0
	CCSA2	5.5726e-08	2.8665e-06	4.1129e-06	100
	GA	4.8186e-01	4.8545e+00	3.0154e+00	0
	PSO	1.1988e-08	1.2401e-06	2.9661e-06	100
FN2	CSA	9.9498e+00	3.1341e+01	1.0594e+01	0
	CCSA1	1.0080e+01	2.6797e+01	9.5278e+00	0
	CCSA2	1.5987e-13	2.9201e-07	5.8813e-07	100
	GA	7.3772e-01	4.2860e+00	3.0752e+00	0
	PSO	9.3746e-09	1.6082e-06	4.5059e-06	100
FN3	CSA	5.9988e-01	8.1558e-01	1.3851e-01	0
	CCSA1	5.9987e-01	7.6367e-01	9.1210e-02	0
	CCSA2	3.4017e-08	7.3404e-05	1.9093e-04	83
	GA	3.5998e+00	7.9732e+00	2.9333e+00	0
	PSO	2.9987e-01	5.5027e-01	9.7618e-02	0
FN4	CSA	2.3749e+00	4.5256e+00	1.5065e+00	0
	CCSA1	9.2891e-01	2.6840e+00	1.1454e+00	0
	CCSA2	2.3161e-08	2.6786e-05	5.5441e-05	93
	GA	5.3800e+02	4.2210e+03	3.9503e+03	0
	PSO	0	1.0000e-01	3.0512e-01	90
FN5	CSA	3.8696e-01	1.8096e+00	6.1417e-01	0
	CCSA1	3.1899e-01	1.2535e+00	6.1896e-01	0
	CCSA2	4.9710e-08	9.6788e-05	1.1192e-04	66
	GA	9.5678e+02	5.2476e+03	3.8364e+03	0
	PSO	1.7748e-11	2.4753e-09	4.6730e-09	100
FN6	CSA	2.5219e-04	1.0043e-03	5.6326e-04	0
	CCSA1	5.3202e-06	1.9262e-05	9.8624e-06	100
	CCSA2	5.8709e-12	4.1725e-06	4.1725e-06	100
	GA	1.6703e+04	3.5632e+04	1.2143e+04	0
	PSO	2.0321e+01	7.4202e+01	3.8643e+01	0
FN7	CSA	1.1276272e-01	2.5119258e-01	8.4614647e-02	0
	CCSA1	9.7187e-02	3.3588e-01	1.0753e-01	0
	CCSA2	1.1102e-16	2.2149e-10	5.5167e-10	100
	GA	1.8128889e-01	2.9271829e-01	5.5367637e-02	0
	PSO	2.0708457e-01	3.3013229e-01	7.1905942e-02	0

Table XII Comparison Of Csa, Ccsa1, Ccsa2, Ga, And Pso For Variant Test Functions With 50d

FN#	Method	Best	Mean	Std.	SR
FN1	CSA	2.1652e-01	4.3313e+00	3.5774e+00	0
	CCSA1	4.0369e-02	4.1700e+00	3.7788e+00	0
	CCSA2	9.4178e-08	9.3449e-06	9.9252e-06	100
	GA	2.9810e+00	1.1677e+01	6.6219e+00	0
	PSO	2.6793e-04	4.624e-03	9.2579e-03	0
FN2	CSA	2.6940e+01	5.0481e+01	1.5383e+01	0
	CCSA1	2.0206e+01	4.5936e+01	1.3231e+01	0
	CCSA2	0	3.2650e-08	1.5328e-07	100
	GA	4.1504e+01	7.6668e+01	2.3645e+01	0
	PSO	1.9899e+01	4.9283e+01	1.6389e+01	0
FN3	CSA	5.9987e-01	7.8658e-01	1.3344e-01	0
	CCSA1	1.1998e+00	1.5232e+00	1.7258e-01	0
	CCSA2	6.3946e-09	1.7673e-05	2.3912e-05	96
	GA	6.8998e+00	1.1099e+01	2.3909e+00	0
	PSO	8.9987e-01	1.1136e+00	1.1702e-01	0
FN4	CSA	1.9576e+00	3.9223e+00	1.0768e+00	0
	CCSA1	4.2684e+00	6.4949e+00	1.2175e+00	0
	CCSA2	5.5298e-08	1.2829e-05	1.8905e-05	100
	GA	2.8330e+03	1.1529e+04	7.0207e+03	0
	PSO	0	2.8666e+00	2.1452e+00	06
FN5	CSA	6.4199e-01	1.8288e+00	9.0756e-01	0
	CCSA1	2.3732e+00	4.3141e+00	1.4229e+00	0
	CCSA2	2.6513e-07	1.7501e-04	2.0145e-04	50
	GA	3.2809e+03	1.1219e+04	5.5946e+03	0
	PSO	2.2420e-04	9.0778e-03	3.5689e-02	0
FN6	CSA	2.9571e-04	8.9627e-04	8.6258e-04	0
	CCSA1	6.5224e-03	1.0983e-05	1.0676e-02	13
	CCSA2	9.6058e-12	3.4821e-08	6.8045e-08	100
	GA	5.4759e+04	1.0150e+05	3.6623e+04	0
	PSO	1.1511e+03	2.5010e+03	8.3540e+02	0
FN7	CSA	1.2525386e-01	2.6059394e-01	8.9133529e-02	0
	CCSA1	1.2005e-01	2.2944e-01	7.4348e-02	0
	CCSA2	0	2.2040e-11	7.3236e-11	100
	GA	1.1399892e-01	2.4195266e-01	5.8161545e-02	0
	PSO	1.0624135e-01	2.3113446e-01	6.8488975e-02	0

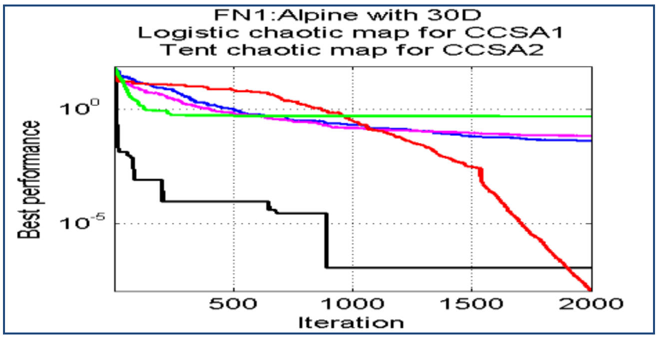

Fig. 3. Convergence curve of FN1 with 30D

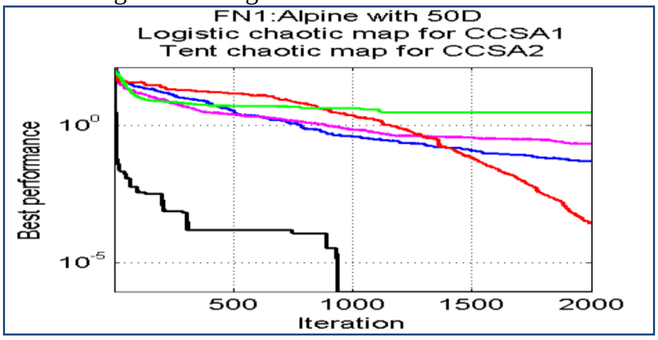

Fig. 4. Convergence curve of FN1 with 50D

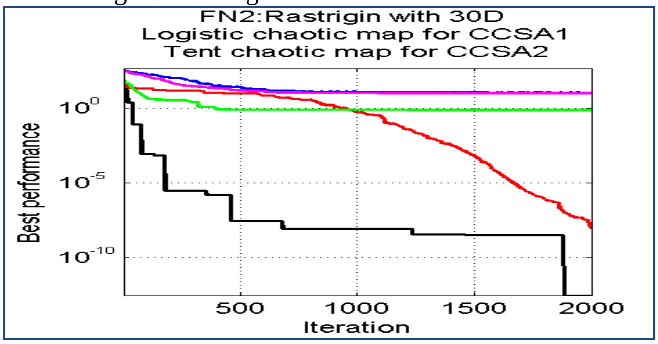

Fig. 5. Convergence curve of FN2 with 30D

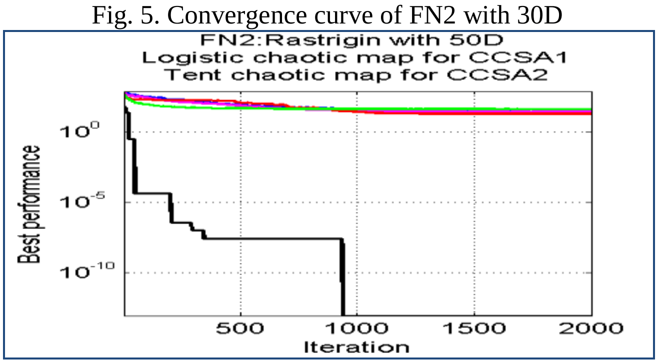

Fig. 6. Convergence Curve Of FN2 With 50D

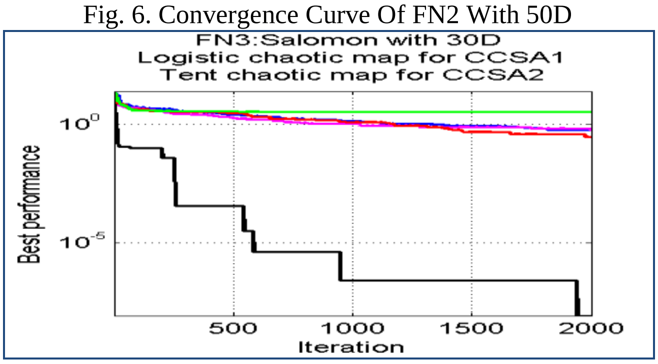

Fig. 7. Convergence curve of FN3 with 30D

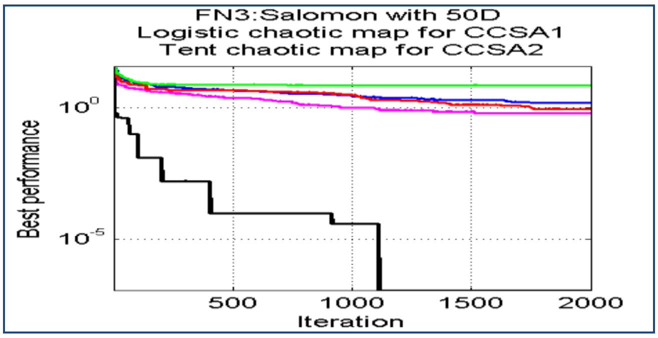

Fig. 8. Convergence curve of FN3 with 50D

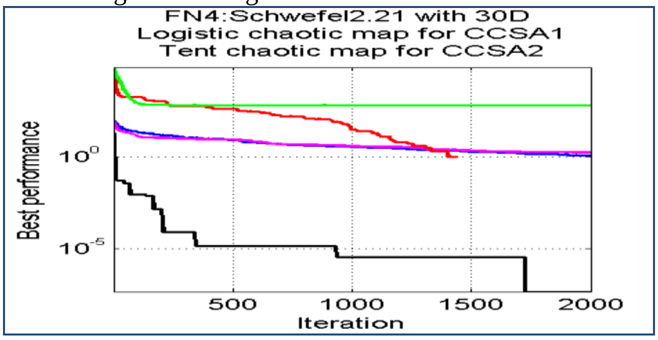

Fig. 9. Convergence curve of FN4 with 30D

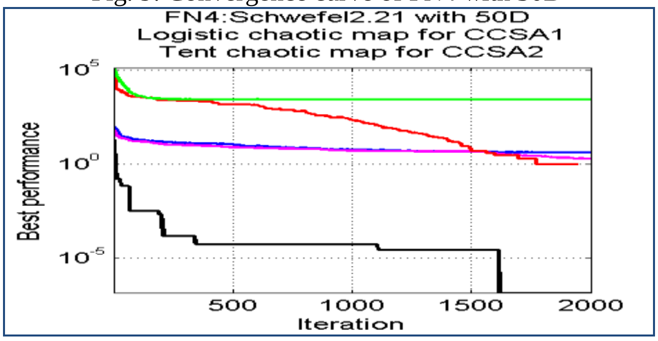

Fig. 10. Convergence curve of FN4 with 50D

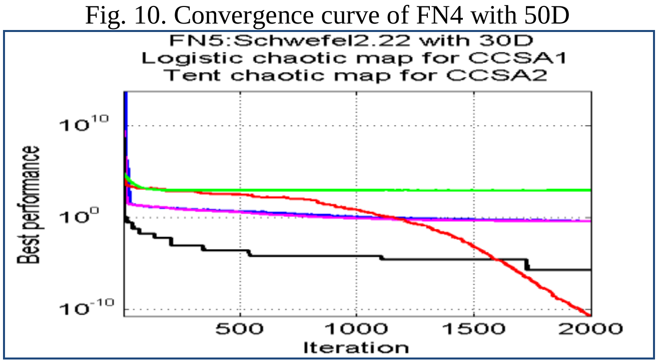

Fig. 11. Convergence curve of FN5 with 30D

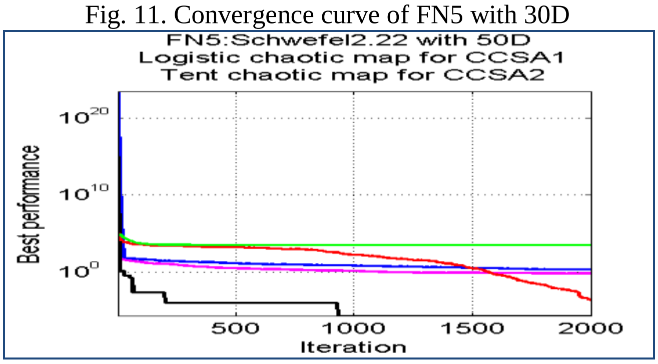

Fig. 12. Convergence curve of FN5 with 50D

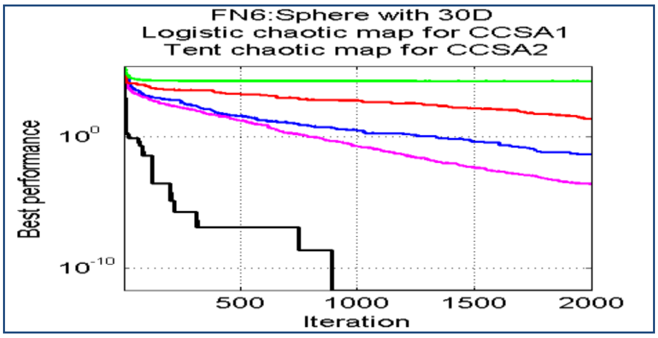

Fig. 13. Convergence curve of FN6 with 30D

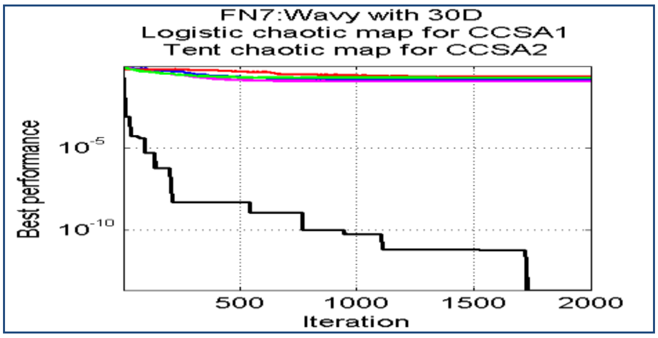

Fig. 15. Convergence curve of FN7 with 30D

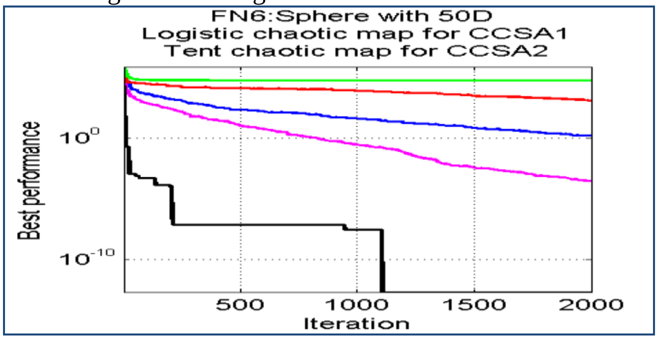

Fig. 14. Convergence curve of FN6 with 50D

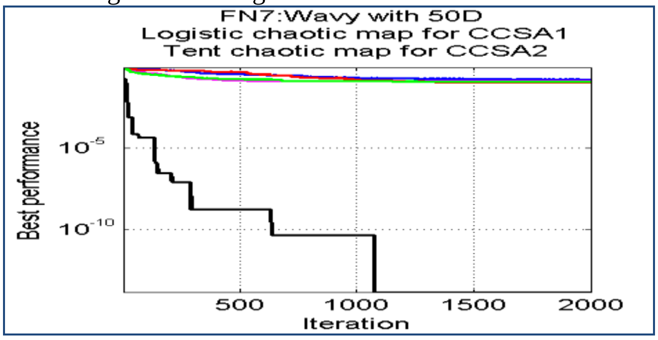

Fig. 16. Convergence

Curve of FN7 with 50D

VIII. REFERENCES

[1] Shilpa S. , and Shyam L., " Multilevel thresholding based on chaotic Darwinian particle swarm optimization for segmentation of satellite images," Applied Soft Computing, vol. 55, pp. 503-522,2017

[2] A. Kaveh, Chaos embedded metaheuristic algorithms, in: Advances in metaheuristic algorithms for the optimal design of structures, Springer, pp. 369-391,2016.

[3] Y. Labbi, D. B. Attous, H. A. Gabbar, B. Mahdad, and A. Zidan, "A new rooted tree optimization algorithm for economic dispatch with value-point effect," Electrical Power and Energy Systems, vol. 79, pp. 298-311,2016.

[4] X. S. Yang, Nature-Inspired Optimization algorithms, Elsevier,2014.

[5] Petra S., and Frantisek Z., "Genetic algorithm using the theory of chaos,"Procedia Computer Science, vol. 51,pp. 316-325,2015.

[6] J. Kennedy and R. Eberhart, "Particle swarm optimization, "In Proc. of IEEE International Conference on Neural Networks, vol. 4, pp.1942-1948, 1995.

[7] Wang GG., Deb S., and Coelho L, "Elephant herding optimization," Computational and Business Intelligence(ISCBI), 2015 3rd International Symposium on. IEEE,,pp. 1-5,2015.

[8] Wang GG., Deb S. and Cui, "Monarch butterfly optimization," Neural Comput & Applic, pp. 1-20,2015.

[9] Najmeh S, Jafar A., and Salwani A., "Kidney-inspired algorithm for optimization problems,"Commun. In Nonlinear Sci. and Numer. Simul., vol. 42,pp. 358-369,2017.

[10] X.S.Yang, "A new metaheuristic bat inspired algorithm," In: Proc.of the workshop on nature inspired cooperative strategies for optimization (NICSO2010). Springer, pp. 65–74.

[11] X.S.Yang,"Firefly algorithm, stochastic test functions and design optimisation, "Int J Bio-Inspired Comput., vol. 2,pp.78–84,2010.

[12] A.H.Gandomi, andA.H.Alavi,"Krill Herd: a new bioinspired optimization algorithm, "Commun Nonlinear Sci Numer Simul, vol. 17(12), pp. 4831-4845, 2012.

[13] A. Askarzadeh, "A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm, " Computers and Structures, vol. 169, pp.1-12, 2016.

[14] Guo P, and Yuming X., "Chaotic glowworm swarm optimization algorithm based on gauss mutation ," In Natural Computation, Fuzzy Systems and Knowledge Discovery(ICNC-FSKD),pp. 205-210,2016, IEEE.

[15] D. Yang, G. Li, and G. Cheng, "On the efficiency of chaos optimization algorithms for global optimization, " Chaos Solution & Fractal ,Vol.34 ,pp. 1366-1375,2007.

[16] A.H. Gandomi, X.S. Yang, S. Talatahari, and A.H. Alavi, "Firefly algorithm with chaos, "Commun. in Nonlinear Sci. and Numer. Simul., vol. 18 (1) , pp. 89-98,2013.

[17] M. Saleh, and M. Haeri, "Comparision of different one-dimensional maps as chaotic search pattern in chaos optimization algorithms, "Applied Mathematics and Computation, vol. 187,pp. 1076-1085,2007.

[18] G. Gharooni-fard, F. Moein-darbari, H. Deldari, and A. Morvaridi, "Scheduling of scientific workflows using a

chaos-genetic algorithm, "Procedia Computer Science, vol. 1, pp. 1445-1454,2010.
[19] A.H. Gandomi, G.J. Yun, X.S. Yang, S. Talatahari, "Chaos-enhanced accelerated particle swarm algorithm, " Communications in Nonlinear Science and Numerical Simulation, vol. 18 (2) ,pp. 327-340,2013.
[20] J. Yi, X. Li, C. H. Chu, and L. Gao, "Parallel chaotic local search enhanced harmony search algorithm for engineering design optimization," Journal of Intelligent Manufacturing, pp. 1-24,2016.
[21] Jun W., Bihua Z., and Shudao Z., "An improved cuckoo search optimization algorithm for the problem of chaotic systems parameter estimation," Computational Intelligence and Neuroscience, vol. 2016, Article ID 2959370, 8 pages, 2016
[22] Seyedali M., and Amir H., " Chaotic gravitational constants for the gravitational search algorithm," Applied Soft Computing, vol. 53 ,pp. 407-419,2017.
[23]S. Talatahari, R. Sheikholeslami, B. Farahmand Azar, and A.H.Gandomi, "Imperialist competitive algorithm combined with chaos for global optimization, " Commun Nonlinear Sci Numer Simul, vol. 17 (3),pp. 1312-1319, 2012.
[24] A. Rezaee Jordehi, "Chaotic bat swarm optimization (CBSO), "Applied Soft Computing, vol. 26,pp.523-530, 2015.
[25] Sayed, G. I., Hassanien, A.E. & Azar, A. T. Neural Comput & Applic(2017). Doi:10.1007/s00521-017-2988-6
[26] H. Afrabandpey, M. Ghaffari, A. Mirzaei, and M. Safayani, " A novel bat algorithm based on chaos for optimization tasks, "In:2014 Iranian Conference on Intelligent Sysems(ICIS), IEEE, pp.1-6, 2014.
[27] G. G. Wang, L. Guo, A. H. Gandomi, G. S Hao, and H. Wang,″Chaotic Krill Herd algorithm, " Information Sciences, vol. 274, pp. 17-34,2014.
[28] M. Mitic, N. Vukovic, M. Petrovic, and Z. Miljkovic, "Chaotic fruit fly optimization algorithm,″Knowl.-Based Syst., vol. 89, pp. 446-458,2015.
[29] M. Jamil and X. S. Yang, "A literature survey of benchmark functions for global optimization problems, "International Journal of Mathematical Modeling and Numerical Optimisation, vol. 4(2),pp.150-194,2013.