Bitlis Eren Univ J Sci & Technol
5 (1), 11 - 17, 2015

E - ISSN 2146-7706

Research Article

Artificial Cooperative Search Algorithm for Parameter Identification of Chaotic Systems

Oguz Emrah Turgut [1], Mert Sinan Turgut [2*], Mustafa Turhan Coban[1]

[1] *Ege University, Faculty of Engineering, Department of Mechanical Engineering, İzmir - Turkey*
[2] *Dokuz Eylül University, Faculty of Engineering, Department of Mechanical Engineering, İzmir - Turkey*
[*] *Corresponding author: sinanturgut@me.com*

Abstract

Parameter estimation of chaotic systems is a challenging and critical topic in nonlinear science. Problem at hand is multi-dimensional and highly nonlinear thereof conventional optimization methods generally fail to extract the unknown parameters of chaotic system. In this study, Artificial Cooperative Search algorithm is put into practice for successful parameter estimation of chaotic systems and compared the parameter estimation performance of Artificial Cooperative Search with Bat, Artificial Bee Colony, Quantum behaved Particle Swarm Optimization algorithms. Parameter identification performance of each algorithm is outlined and benchmarked with several numerical simulations including Lörenz system, Duffing equation and Josephson junction. Results show that Artificial Cooperative Search algorithm outperforms other algorithms in terms of robustness and effectiveness.

Keywords: Artificial cooperative search, Chaotic systems, Metaheuristic algorithms, Parameter identification.

1. Introduction

Chaos is a dynamic, unpredictable and complex phenomenon which occurs in nonlinear systems. It shows an unstable behaviour which is very sensitive to initial conditions and involves infinite unstable periodic motions (Wang et. al. 2011). Many natural and simulated phenomena such as voice generation, earthquakes, laser systems and epileptic seizures serve chaotic behaviour (Chang et. al. 2008). All these events were thought to be stochastic and even unpredictable however; time series generated by the chaotic systems can be predicted if mathematical models of chaotic systems are successfully constructed.

Chaos has been applied to various areas such as chemical reactions, power converters, biological systems, information processing, secure communication, and economics (Wang & Xu 2011). Many mathematical models and simulations have been carried out for controlling and synchronization of chaotic systems during two decades (He et al 2007; Tang & Guan 2009; Modares et al. 2010). Such models have been applied in definite chaotic systems with predetermined system parameters however; there generally exist parameter mismatches and distortions in real world problems (Wang et al 2010). Therefore, this topic has become popular among the researchers and numerous scientific studies have been proposed to overcome this drawback by suggesting novel solution strategies (Rahul 2005; Peng et al. 2009; Biswambhar et al. 2011).

Parameter estimation of chaotic system is converted to parameter optimization by virtue of constructing appropriate fitness function. Metaheuristic algorithms have commonly been utilized for estimating the parameters of chaotic systems since they are derivative free and do not require domain information. Genetic algorithms (Dai et. al. 2002), Particle Swarm Optimization (Tang & Guan 2009; Ko et al. 2010; Sun et al. 2010) Bee Colony Algorithm (Gholipour et al. 2013), Invasive Weed Optimization (Ahmadi & Mojallahi 2012) , Firefly algorithm (Gao et al. 2013), Ant Swarm algorithm (Li et al. 2006), Cuckoo Search algorithm (Xiang-Tao & Ming-Hao 2012), Harmony search (Coelho & Bernert 2009), Gravitational Search (Li et al. 2012) and some of the hybrid algorithms (Tien & Li 2012; Wang & Li 2012) were utilized for parameter estimation of chaotic systems and most of them succeeded in finding true values of system parameters however, there is still room to improve the best results obtained by these algorithms. In this study, Artificial Cooperative Search (ACS) (Civicioglu 2013) algorithm is suggested for parameter identification of chaotic systems. To the best of author's knowledge, Artificial Cooperative Search algorithm has not been utilized in estimating parameters of chaotic systems yet. Performance of ACS algorithm will be compared with Quantum behaved Particle Swarm Optimization (QPSO) (Sun et al. 2004), Artificial Bee Colony (ABC) (Karaboga 2005), Bat Algorithm (BAT) (Yang et al. 2010) in terms of best, worst, mean and standard deviation values. To upgrade the search mechanism of the algorithms, randomized algorithm parameters are used which are bounded between real valued numbers instead of static parameters. Effectiveness of the optimizers is compared by means of applying numerical simulations based on chaotic systems including Lorenz system, Duffing equation and Josephsen junctions.

Rest of the paper is organized as follows: Section 2 describes the parameter estimation of chaotic differential systems in optimization point of view. Section 3 gives the detailed description Artificial Cooperative Search algorithm. Numerical simulations, comparisons and discussions are performed in Section 4. Finally, the article is concluded with some remarkable comments in Section 5.

Received 24 February 2015
Available online June 2015

2. Formulation of the problem

Parameter identification of chaotic systems will be clarified in this section. Consider n - dimensional continuous nonlinear chaotic differential equation system:

$$\dot{X} = G\left(X, X_0, \theta\right) \tag{1}$$

where $X = \left(x_1, x_2, \ldots, x_n\right)^T \in R^n$ the state is vector; X_0 specifies the initial state and $\theta = \left(\theta_1, \theta_2, \ldots, \theta_m\right)^T \in R^m$ is the set of system parameters; $G: R^n \times R^m \to R^n$ is a nonlinear vector function. Presuming that the nature of the system is known in advance, parameter estimation system can be stated as the following

$$\dot{\hat{X}} = G\left(\hat{X}, \hat{X}_0, \hat{\theta}\right) \tag{2}$$

where $\hat{X} = \left(\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_n\right)^T \in R^n$ is the estimated state vector; $\hat{X}_0$ is estimated initial state vector; $\hat{\theta} = \left(\hat{\theta}_1, \hat{\theta}_2, \ldots, \hat{\theta}_m\right)^T \in R^m$ is the estimated parameters of chaotic system. Parameter estimation problem can be defined as

$$min\ J = \sum_{k=1}^{W} \left\| X_k - \hat{X}_k \right\| \tag{3}$$

where W is the length of state variables; X_k and $\hat{X}_k$ are the measurable state variables of original and estimated system at time $k = 1, 2, \ldots, W$. Optimization objective is to estimate parameters of system while minimizing fitness function J. Parameter estimation of chaotic system is a hard and tedious task since it is a multi-dimensional nonlinear optimization problem. In addition, there are many local optima in search space due to the multidimensionality of objective function and this makes it very difficult to obtain global best solution with traditional optimization algorithms. Therefore, in this study, Artificial Cooperative Search algorithm is applied in order to estimate the unknown parameters of chaotic systems. Framework of parameter identification in the view of optimization can be depicted in Figure 1.

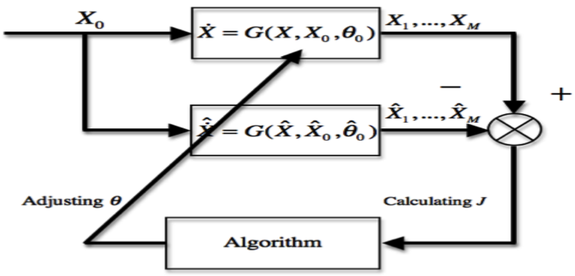

Figure 1. The principle of parameter identification for a chaotic system

3. Artificial Cooperative Search

ACS is swarm intelligence based metaheuristic algorithm used for solving numerical optimization problems. The algorithm is based on interaction between two artificial superorganisms as they interact and migrate to different zones to find global minimum of a problem.

Amount of food that can be found in a specific zone is tied to yearly climate a change. For this reason, superorganisms develop some kind of seasonal migration behaviour to discover better food sources. It is known that, in nature, most species form superorganisms and divide into sub-groups (sub-superorganisms) prior to migration in order to find a better food source. This superorganism behaviour is determined by the coordination of sub-groups.

Interaction and explorer usage are two main behaviours of the superorganisms. Before migrating to a new zone, first, superorganism sends an explorer to collect information about the possible migration zones. Then, explorer shares the information with the superorganism individuals and these individuals give their opinion for the possibility of migration to the new discovered areas. During the migration process, exploration behaviour is sustained to find better zones. Interaction behaviour is another important behaviour among the living species. All superorganisms living in the same habitat, naturally interacts with each other. Parasite/host or predator/prey relationships may emerge in alturation, coextinction, coevolution or cooperation interactions between superorganisms.

In ACS algorithm, two superorganisms named α and β consist of random solutions of the problem move to more fruitful nesting or feeding areas. Each superorganism consists of N members and each sub-superorganism consists of D members, which corresponds to dimension of the problem. Also, the two superorganisms decide the predator and prey sub-superorganisms. Predator sub-superorganism tracks the prey sub-superorganism while they move towards global minimum of the problem.

The initial values of the individuals of the two superorganisms are calculated by using Equation (4).

$$\begin{aligned} \alpha_{i,j,t} &= rnd \times (up_j - low_j) + low_j \\ \beta_{i,j,t} &= rnd \times (up_j - low_j) + low_j \end{aligned} \tag{4}$$

where $i = 1, 2, 3, \ldots, N$, $j = 1, 2, 3, \ldots, D$ and $t = 1, 2, 3, \ldots, \mathrm{maxiter}$. The g value represents the iteration number while rnd represents a random number chosen from a uniform distribution between [0, 1]. up_j and low_j represents the upper and lower bounds of the search space for j^{th} dimension of the problem. Fitness values of the associated sub-superorganisms are determined by using Equation (5).

$$\begin{aligned} y_{i;\alpha} &= f(\alpha_i) \\ y_{i;\beta} &= f(\beta_i) \end{aligned} \tag{5}$$

Predator individuals are determined by the rule in Algorithm 1.

Algorithm 1 Calculation of Predator individuals

if $rnd < rnd$
  Predator = α, $y_{\text{Predator}} = y_a$, key = 1
else
  Predator = β, $y_{\text{Predator}} = y_\beta$, key = 2
end

Prey individuals are determined by the rule in Algorithm 2.

Algorithm 2 Calculation of Prey individuals

if $rnd < rnd$ Prey = α *else* Prey = β *end*
Prey = permute (Prey)

In Algorithm 2, permute() function randomly changes the places of all row elements of prey individuals. Since only active individuals are permitted to migrate, passive individuals are determined by Algorithm 3.

Algorithm 3 Calculation of passive individuals by binary valued integer map (M)

$M_{N\ x\ D} = 1$
for all elements in M
  $if\ \ rnd < (p \times rnd)\ \ then\ \ M_{rndint(N),\ rndint(D)} = 0\ \ \ end$
end
$if\ \ rnd < (p \times rnd)\ \ then$
  for i=1 to N
    for j=1 to D
      $if\ \ rnd < (p \times rnd)\ then\ \ M_{i,j} = 1\ \ else\ \ M_{i,j} = 0\ \ end$
    end
  end
end
for i=1 to N
  $if\ \sum_{j=1}^{D} M_i = D\ \ then\ \ M_{i,rndint(D)} = 0\ \ end$
end

rndint() function generates random integers between a chosen interval by employing gauss distribution and p represents the probability of biological interaction. Biological interaction location between prey and predator individuals is calculated by using the following equation:

$$x = Predator + R \times (Prey\text{ - }Predator) \tag{6}$$

R is a variable that controls the speed of biological interaction. R is generated by using the procedure in Algorithm 4.

Algorithm 4 Decision rule to obtain scale factor (R)

$if\ \ rnd < rnd\ \ then$
  $R = 4 \times rnd \times (rnd - rnd)$
else
  $R = \Gamma(4 \times rnd, 1)$
end

where rnd is a random number between 0.0 and 1.0; Γ() represents the gamma distribution with a shape parameter of $4 \times rnd$ and scale parameter of 1.0. Position update procedure used by active individuals is shown in Algorithm 5.

Algorithm 5 Updating of biological interaction locations by active individuals

for i=1 to N
  for j=1 to D
    $if\ \ M_{i,j} > 0\ then\ \ x_{i,j} = Predator_{i,j}\ \ \ end$
  end
end

If biological interaction locations exceed boundaries, new locations are generated according to the rule in Algorithm 6.

Algorithm 6 Application of boundary control mechanism

for i=1 to N
  for j=1 to D
    $if\ \left(x_{i,j} < low_j\right) \vee \left(x_{i,j} > up_j\right) then$
      $x_{i,j} = low_j + rnd \times \left(up_j - low_j\right)$
    end
  end
end

Predator sub-superorganisms are compared with biological interaction locations concerning their fitness values. If the fitness values of biological locations are better than objective function values of Predator individuals, Predator individuals are updated by implementing the methodology given in Algorithm 7.

Algorithm 7 Predator sub-superorganism update

for i=1 to N
  $if\ \ f(x_i) < y_{i,Predator}\ \ then$
    $Predator_i = x_i$
    $y_{i,Predator} = f(x_i)$
  end
end

New α and β superorganisms and their fitness values for next generations are determined by the strategy presented in Algorithm 8 with the utilization of "key" parameter decided in Algorithm 1.

Algorithm 8 Determination of new sub superorganisms for next generations

if key=1 then
  $\alpha = Predator,\ y_\alpha = y_{Predator}$
else
  $\beta = Predator,\ y_\beta = y_{Predator}$
end

Pseudocode of Artificial Cooperative Search algorithm is presented in Table 1.

Table 1. Psudocode of Artificial Cooperative Search

Initialize population size, problem dimension, maximum number of iteration, lower and upper bounds,initialize Superorganisms (α , β) and determine their corresponding fitness values with Equation (4) and (5)

for iter = 1 to maxiter
  Calculate Predator individuals by applying Algorithm 1
  Determine Prey individuals with Algorithm 2
  Calculate the scale factor (R) with Algorithm 4
  Calculate passive individuals by binary valued integer map as explained
  in Algorithm 3
  Decide biological interaction locations with Equation 3

Update biological interaction locations by active individuals with
the rules of Algorithm 5
Apply boundary control with Algorithm 6
Update Predator superorganisms by Algorithm 7
Determine new superorganisms for next generations by Algorithm 8
Store the best solution and its corresponding fitness value
end
Output best results

4. Simulations and results

This section gives the performance comparison of metaheuristic algorithms on extracting of unknown parameters of chaotic systems including Lorenz system, Duffing system, and Josephson junctions. Numerical simulations are performed in Java™ executing Pentium i5 CPU @ 2.5 GHz and 6.0 GB RAM on personal computer. Sampling intervals (h) and total sampling points (W) for each chaotic system are set to h=0.01 and W=200 respectively. Fourth order adaptive Runge - Kutta method is applied for solving system of differential equations which are selected for testing the effectiveness of discussed metaheuristic algorithms. For the sake of comparison reliability, upper and lower limits of estimated parameters are taken all same for each algorithm. In addition, maximum generation (iteration) and population size of each algorithm are fixed to 5000 and 20 respectively. Algorithms are run for 100 times due to the stochastic discrepancy. For Artificial Bee Colony algorithm, limited food source is set to 2000. For Quantum behaved Particle Swarm Optimization, cognitive and social learning factors are set to real valued numbers that are randomly generated between 0.0 and 3.0. For Bat Algorithm, α and β values are set to uniform random numbers between 0 and 1; f_{min} and f_{max} are fixed to 0.0 and 2.0, respectively; γ is a random number between 0.0 and 5.0; initial values of loudness of the sound pulse (A) and pulse emission rate (r) are selected as uniformly distributed random numbers between 0.0 and 1.0. For each algorithm, best solutions are taken into consideration while plotting evolution of fitness function and system parameters.

4.1. Lorenz System

First example is employed as Lorenz system (Zhou et al. 2004) which is formulated as follows

$$
\begin{aligned}
\dot{x}_1 &= a\left(x_2 - x_1\right) \\
\dot{x}_2 &= bx_1 - x_1x_3 - x_2 \\
\dot{x}_3 &= x_1x_2 - cx_3
\end{aligned}
\tag{7}
$$

where x_1, x_2 and x_3 are the state variables; a, b, and c are unknown algorithm parameters. This system serves chaotic behaviour when a = 10, b = 28 and c = 8/3. Initial values of state variables are $\vec{x}^0 = \left[x_1^0, x_2^0, x_3^0\right] = [1.0, 1.0, 1.0]$.

Search space of a, b and c parameters are restricted between [5, 15], [20, 30] and [0, 5], correspondingly. Table 2 reports the statistical results of parameter identification of Lorenz system. Table 2 shows that ACS and QPSO finds true values of system parameters however, ACS algorithm is superior to all algorithms in terms of robustness and even worst results obtained by ACS algorithm are better than BAT, ABC algorithm.

Table 2. Statistical results of different algorithms for Lorenz system

		ACS	BAT	ABC	QPSO
BEST	a	10.00000	9.99998	10.0003	10.00000
	b	28.00000	27.9998	27.9999	28.00000
	c	2.66666	2.6666	2.6666	2.66666
	J_{min}	0.000000	0.00676	0.19198	0.000000
AVG.	a	10.00000	9.99997	10.00225	10.03121
	b	28.00000	28.0000	27.99594	27.95318
	c	2.66666	2.6666	2.66626	2.66228
	J_{avg}	2.18E-13	0.04285	3.47033	26.09139
WORST	a	9.99999	9.9999	10.01553	10.62163
	b	28.00000	27.9999	27.97346	27.09904
	c	2.66666	2.6666	2.66379	2.58054
	J_{max}	6.52E-12	0.09356	8.771272	300.2061

Figure 2(a) shows the convergence properties of fitness values for all mentioned algorithms.

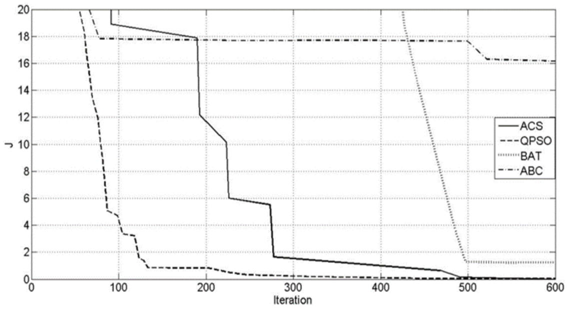

Figure 2(a). Convergence history of fitness function for each algorithm for Lorenz system

In Figure 2(a), it is seen that QPSO algorithm reaches optimum solution after 247 iterations whereas ACS finds optimum solution after 497 iterations. BAT and ABC algorithms have not obtained optimum solution within 600 iterations. Figure 2(b)-(d) depicts the convergence process of equation parameters for each algorithm in a single run generating the best solution.

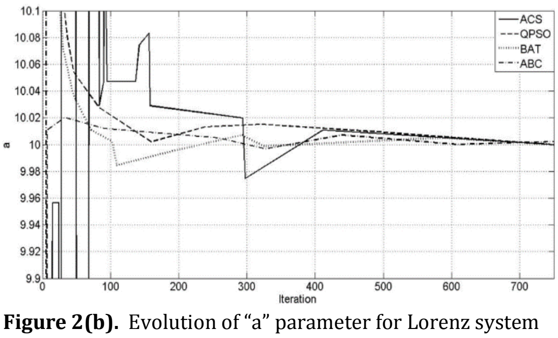

Figure 2(b). Evolution of “a” parameter for Lorenz system

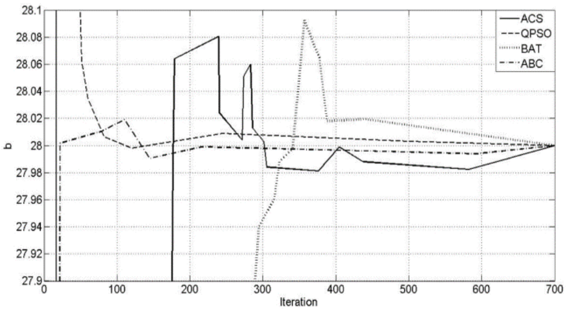

Figure 2(c). Evolution of “b” parameter for Lorenz system

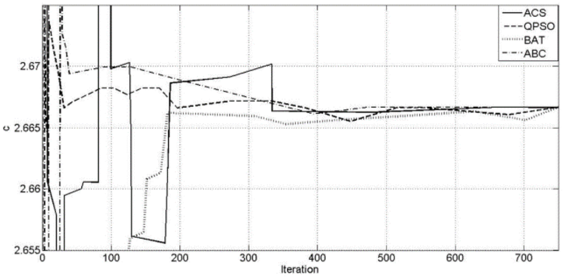

Figure 2(d). Evolution of "c" parameter for Lorenz system

4.2. Duffing equation

Duffing equation (Yang et al. 2009) is a non-linear second order differential equation which simulates damped and driven oscillators. This equation generates chaotic sequences when *a* = 4.0, *b* = 1.1 and *c* = 1.0 and given as:

$$\ddot{x} + a\dot{x} - bx + cx^3 = 2.1sin(1.8t) \tag{8}$$

Initial state of system is $x_0 = 0.1,\ \dot{x}_0 = 0.1$. Upper and lower bounds for algorithm parameters are defined as $0.0 \le a \le 20.0$, $0.0 \le b \le 10.0$, and $0.0 \le c \le 10.0$. For Duffing equation, statistical analysis is given in Table 3.

Table 3. Statistical results of different algorithms for Duffing equation

		ACS	BAT	ABC	QPSO
BEST	*a*	0.39999	0.39943	0.39803	0.39999
	b	1.09999	1.09885	1.09556	1.09999
	c	1.00000	0.99979	0.99908	0.99999
	J_{min}	0.00000	3.15E-4	0.01307	6.79E-6
AVG.	*a*	0.39999	0.44065	0.40532	0.43922
	b	1.09999	1.13315	1.11219	1.18903
	c	1.00000	1.03700	1.00249	1.01819
	J_{avg}	9.8E-14	0.64426	0.15462	0.73188
WORST	*a*	0.39999	1.50000	0.28742	0.03964
	b	1.09999	2.00000	0.84527	0.27853
	c	1.00000	2.00000	0.94766	0.83003
	J_{max}	4.6E-13	17.3583	0.76635	2.58151

Comparisons reveal that ACS algorithm outperforms other algorithms concerning robustness and best solution. Figure 3(a) depicts the evolving history of fitness values and unknown parameters of algorithms.

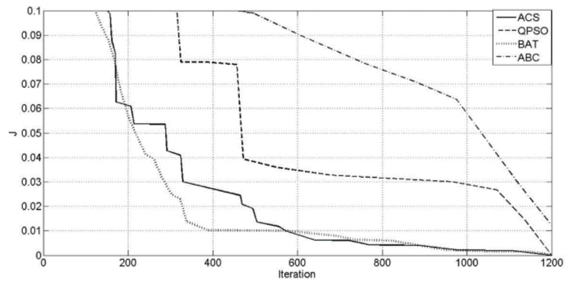

Figure 3(a). Convergence history of fitness function for each algorithm for Duffing equation

Figure 3(a) presents that ACS finds optimum solution after 914 iterations while other algorithms have not reached the optimum within 1200 iterations. Figure 3(b)-(d) shows the convergence performance of algorithm parameters for each algorithm. As it is clarified in Figure 3(b)-(d), ACS converges to true values of system parameters more quickly than other algorithms.

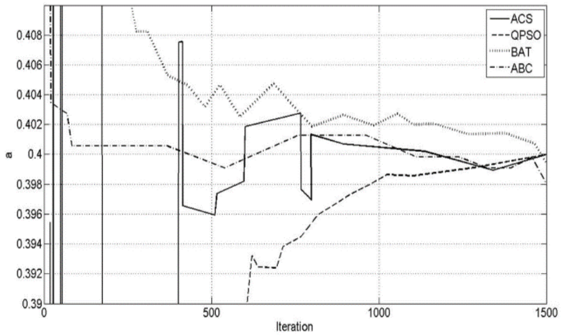

Figure 3(b). Evolution of "a" parameter for Duffing equation

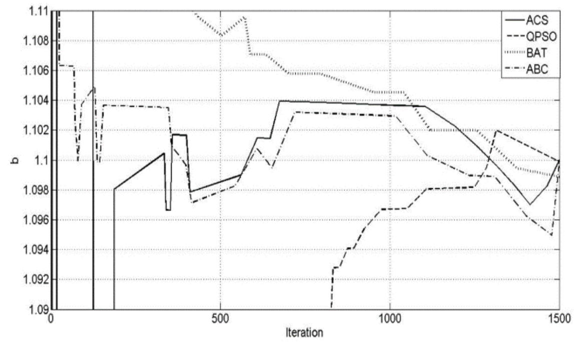

Figure 3(c). Evolution of "b" parameter for Duffing equation

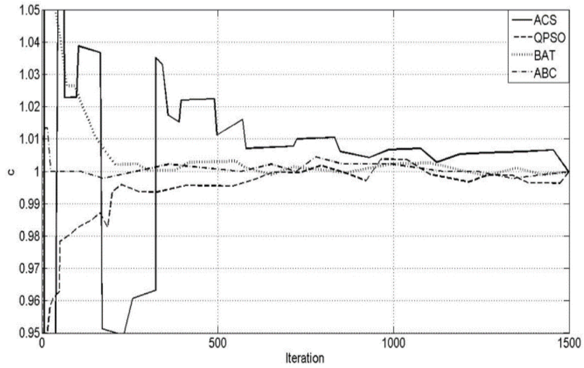

Figure 3(d). Evolution of "c" parameter for Duffing equation

4.3. Josephson junctions

The study of the chaotic behaviour of (quantum) Josephson junctions (Yeh & Kao 1983) is of much fundamental and even practical interest. Written in dimensionless form, the differential equation for the quantum phase difference, Φ, across the junction is given by

$$\ddot{\Phi} + (\beta_c)^{-0.5}\,\dot{\Phi} + sin\Phi = Asin\Omega t \tag{9}$$

where β_c is the so-called McCumber parameter and *Ω* is the (normalized) angular frequency of the driving current. Equation (11) is in chaotic state when $\beta_c = 4$, *Ω = 0.47* and *A = 0.9045* with the initial conditions of $\Phi = \dot{\Phi} = 0$. Searching ranges of the system parameters are defined as: $0.0 \le \beta_c \le 10.0$, $0.0 \le \Omega \le 5.0$, $0.0 \le A \le 5.0$. Table 4 reports the statistical results for this simulation.

Table 4. Statistical results of different algorithms for Josephson junctions

		ACS	BAT	ABC	QPSO
BEST	a	0.90449	0.90190	0.93973	0.90468
	b	0.49999	0.49863	0.51710	0.50010
	c	0.47000	0.47121	0.45398	0.46991
	J_{min}	6.58E-11	8.79E-5	0.02036	1.89E-4
AVG.	a	0.90450	0.90426	0.86367	0.91661
	b	0.49999	0.49991	0.48020	0.49599
	c	0.46999	0.47011	0.49355	0.47107
	J_{avg}	2.07E-8	2.46E-4	0.07355	0.04458
WORST	a	0.90450	0.90526	0.97797	0.62918
	b	0.49999	0.50034	0.57497	0.27842
	c	0.46999	0.46977	0.44551	0.64321
	J_{max}	3.58E-7	5.15E-4	0.19651	0.24246

In Table 4, ACS finds almost true values of system parameters in every try and shows its superiority over the other algorithms in terms of robustness and solution accuracy. In addition, even the worst solution obtained by ACS is much better than the best solutions acquired by the other algorithms. Figure 4(a) visualizes the evolving process of fitness values for each algorithm.

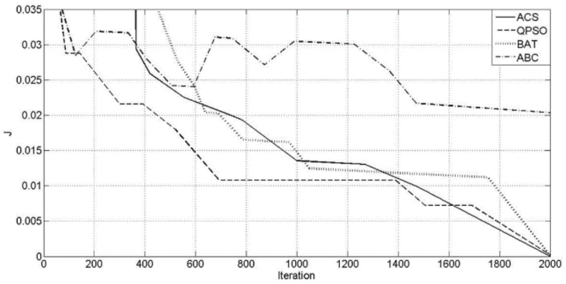

Figure 4(a). Convergence history of fitness function for each algorithm for Josephson junctions

In Figure 4(a), it is shown that ACS converges to optimal solution more quickly and obtains optimum after 1998 iterations. Figure 4(b) to (d) depicts the convergence rates of system parameters and it is observed that ACS converges to true values more rapidly than the others as all system parameters reaches to optimum less than 1500 iterations.

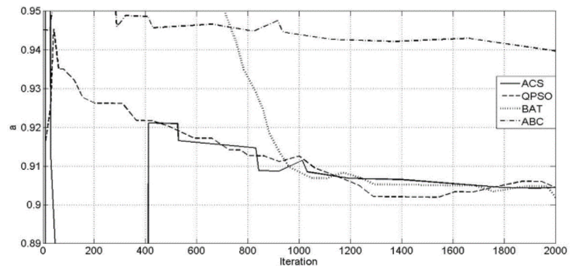

Figure 4(b). Evolution of "a" parameter for Josephson junctions

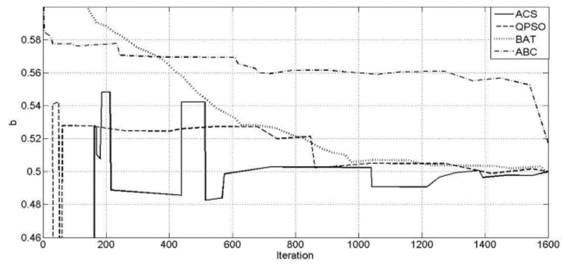

Figure 4(c). Evolution of "b" parameter for Josephson junctions

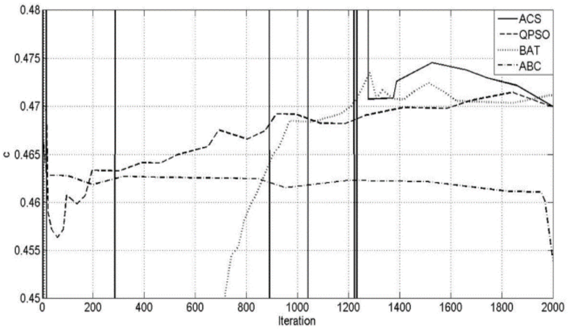

Figure 4(d). Evolution of "c" parameter for Josephson junctions

5. Conclusion

In this study, Artificial Cooperative Search algorithm is introduced to identify the unknown parameters of chaotic systems which are formulated as a multi-dimensional continuous optimization problem. Statistical results those are obtained from ACS are compared with the solutions acquired by Artificial Bee Colony, Quantum behaved Particle Swarm and Bat algorithms. To improve the algorithm performance, randomized algorithm parameters are utilized instead of static parameters. Simulation results of chaotic Lorenz system, Duffing equation and Josephson junctions indicate that Artificial Cooperative Search algorithm has better performance than Artificial Bee Colony, Quantum behaved Particle Swarm and Bat algorithm in identifying system parameters since Artificial Cooperative Search can estimate the system parameters more accurately, more rapidly and more stably. For a future work, Artificial Cooperative Search which gives the best performance for this study will be applied on time-delay systems.

References

Ahmadi M, Mojallali H (2012). Chaotic invasive weed optimization algorithm with application to parameter estimation of chaotic systems. Chaos Soliton Fract 45, 1108–1120

Biswambhar RA, Roy C, Papri S (2011). Parameter identification of a delay dynamical system using synchronization in presence of noise. Chaos Soliton Fract 32(4), 1278–1284.

Chang J, Yang Y, Liao T, Yan J (2008). Parameter identification of chaotic systems using evolutionary programming approach. Expert Syst. Appl. 35, 2074–2079.

Civicioglu P (2013). Artificial cooperative search algorithm for numerical optimization problems. Inform. Sciences. 229, 58–76.

Coelho LS, Bernert DLA (2009). An improved harmony search algorithm for synchronization of discrete-time chaotic systems. Chaos Soliton Fract 41, 2526–2532.

Dai D, Ma XK, Li FC, You Y (2002). An approach of parameter identification for a chaotic system based on genetic algorithm. Acta Phys Sin-Ch Ed. 11, 2459–2462.

Gao W, Zhang Z, Chong Y (2013). Chaotic System Parameter Identification Based on Firefly Optimization. Appl. Mech. Mater. (347-350), 3821-3826

Gholipour R, Khosravi A, Mojallali H (2013). Parameter Estimation of Lorenz Chaotic Dynamic System Using Bees Algorithm. International Journal of Engineering 26(3), 257-262.

He Q, Wang L, Liu B (2007). Parameter estimation for chaotic systems by particle swarm optimization. Chaos Soliton Fract 34, 654–61.

Karaboga D (2005). An idea based on honey bee swarm for numerical optimization, Technical Report-RE06, Erciyes University, Engineering Faculty, Computer Engineering Department.

Ko C, Fu Y, Lee C, Wu C (2010). Parameter estimation of chaotic systems by a nonlinear time-varying evolution PSO method. Artif Life Robotics 15, 33–36.

Li L, Yang Y, Peng H, Wang X (2006). Parameter identification of chaotic systems via chaotic ant swarm. Chaos Soliton Fract 28, 1204–1211.

Li C, Zhou J, Xiao J, Xiao H (2012). Parameters identification of chaotic system by chaotic gravitational search algorithm. Chaos Soliton Fract 45, 539–547.

Modares H, Alfi A, Fateh M (2010). Parameter identification of chaotic dynamics systems through an improved particle swarm optimization. Expert Syst Appl 37, 3714–3720.

Peng B, Liu B, Zhang FY, Wang L (2009). Differential evolution algorithm based parameter identification for chaotic systems. Chaos Soliton Fract 39(5), 2110–2118.

Rahul K (2005). Identification of all model parameters of chaotic systems from discrete scalar time series measurements. Phys Lett A 346 (4), 275–280.

Sun J, Feng B, Xu WB. Particle swarm optimization with particles having quantum behaviour. In: The Congress on Evolutionary Computation, IEEE Press, 2004, Portland.

Sun J, Zhao J, Wu X, Fang W, Cai Y, Xu W (2010). Parameter estimation for chaotic systems with a Drift Particle Swarm Optimization method. Phys Lett A 374, 2816–2822

Tang Y, Guan X (2009). Parameter estimation for time-delay chaotic system by particle swarm optimization. Chaos Soliton Fract 40, 1391–1398.

Tang Y, Guan X (2009). Parameter estimation of chaotic system with time delay: A differential evolution approach. Chaos Soliton Fract 42, 3132–3139.

Tien J, Li TS (2012). Hybrid Taguchi-chaos of multilevel immune and the artificial bee colony algorithm for parameter identification of chaotic systems. Comput Math Appl 64, 1108–1119.

Wang S, Chang Y, Li X, Zhang J (2010). Parameter identification for a class of nonlinear chaotic and hyperchaotic flows. Nonlinear Anal-Theor 11, 423-431.

Wang L, Xu Y (2011). An effective hybrid biogeography-based optimization algorithm for parameter estimation of chaotic systems. Expert Syst Appl 38, 15103–15109.

Wang L, Li L (2010). An effective hybrid quantum-inspired evolutionary algorithm for parameter estimation of chaotic systems. Expert Syst Appl 37, 1279–1285.

Wang L, Xu Y, Li L (2011). Parameter identification of chaotic systems by hybrid Nelder–Mead simplex search and differential evolution algorithm. Expert Syst Appl 38, 3238–3245.

Xiang-Tao L, Ming-Hao Y (2012). Parameter estimation for chaotic systems using the cuckoo search algorithm with an orthogonal learning method. Chin Phys B 21 (5), 050507(1-6)

Yang K, Maginu K, Nomura H (2009). Parameters identification of chaotic systems by quantum-behaved particle swarm optimization. Int J Comput Math 86(12), 2225-2235.

Yang XS, A New Metaheuristic Bat-Inspired Algorithm, in: Nature Inspired Cooperative Strategies for Optimization, NISCO 2010, Studies in Computational Intelligence, Springer Berlin 284, 65–74.

Yeh WJ, Kao YH (1983). Intermittency in Josephson junctions. Appl. Phys. Lett 42, 1599-1602.

Zhou T, Chen G, Tang Y (2004). A universal unfolding of the Lorenz system. Chaos Soliton Fract 20, 979–93