MODELING TIME SERIES FORECASTING USING EVOLUTIONARY TECHNIQUES

Sarabjeet Singh

IKG PTU and Faculty CCET Sec-26, Chandigarh-160019 India

Satvir Singh

Faculty PIT Kapurthala IKG PTU Kapurthala-144603 (Pb) India

Vijay Kumar Banga

Faculty ACET. Amritsar-143109 (Pb) India

ABSTRACT

In recent years, usage of time series forecasting has been increasing day by day for prediction like, share market, weather forecasting and data analysis. Forecasting of Mackey Glass chaotic time series has been carried out in this paper. It is considered that prediction of a chaotic time series system is a nonlinear, multi-variable and multi-modal optimization problem. To get an optimum output of times series, global optimization techniques are required in order to minimize the effect of local optima. Application of recent evolutionary techniques have been considered as pervasive technology for Optimization. In this paper, Fuzzy Logic System (FLS) deals with non-linearity and generates the rule base from training data used for time series forecasting. Further, application of five recent evolutionary techniques have been considered for optimization like Genetic Algorithm (GA) and Gravitational Search Algorithm Particle Swarm Optimization (GSA-PSO),. A comparison for bench mark data of time series forecasting is done using above discussed techniques. it is observed that GA performs better as compared to GSAPSO in both terms, i.e. accuracy and time.

Keywords: Fuzzy Logic System, Time Series Forecasting, Genetic Algorithm, Gravitational Search Algorithm, Particle Swarm Optimization

Cite this Article:. Sarabjeet Singh, Satvir Singh and Vijay Kumar Banga, Modeling Time Series Forecasting using Evolutionary Techniques, *International Journal of Advanced Research in Engineering and Technology, 10*(1), 2019, pp 34-44.
http://www.iaeme.com/IJARET/issues.asp?JType=IJARET&VType=10&IType=1

1. INTRODUCTION

Time series prediction has been attracted researchers since last decades towards non-linear dynamic systems. Forecasting using time series provides remarkable development to forecast future values for dynamical models and series for various applications [1] [2]. However, prediction of such models totally depend upon the experimental behaviour of the real system.

In non-linear, multivariable and multimodal systems parametric estimation is a challenge to get a required output from the system[3].

In many practical applications, FLS has strong ability to deal with non-linearity of the system to provide a linear system. FLS has been successfully implemented to provide precise and significant information from imprecise and inexact information. The ease of translation of human reasoning and concept formation into fuzzy rules is the powerful aspect of FLS. However, selection of type and number of membership functions with scaling factors used in fuzzification and defuzzification are still a bottleneck in designing of FLS as they are generally decided using trial and error process.

In recent years, bio-inspired computational algorithms termed as computation Intelligence (CI) have delivered effective/optimum solution for many applications across the globe. Among various CI techniques like, ANN, FLS and EAs have been integrated for prediction of chaotic time series.

In [4], two computational intelligence techniques, single multiplicative neuron model with co-operative CO-PSO and adaptive neuro-fuzzy inference system have been proposed for time series prediction, authors compared both proposed techniques and observed that ANFIS performs better as compared to COPSO in terms of accuracy and time. Further authors [3] extended the work and proposed teaching learning based optimization with differential evolution (TLBO-DE) algorithm for chaotic time series prediction, made a comparison with PSO and observed that TLBO-DE performs better in terms of efficiency and robustness. In [5], a new model has been proposed to forecast monthly ED visits.

In [6], artificial bee colony is compared with GA, PSO, Particle swarm inspired evolutionary algorithm (PSO-EA) and observed that ABC outperforms other algorithms for selected bench mark optimization functions. Authors in [7] discussed various engineering applications. In [8], method for fuzzy forecasting based on two-factors second-order fuzzy-trend logical relationship groups and particle swarm optimization for time series forecasting has been proposed and authors concluded that the proposed method performs better as compared to other existing methods. In [9], a new hybrid PSO algorithm using gravitational search algorithm and PSO have been presented for function optimization and observed that PSOGSA performs better as compared to PSO and GSA for function optimization with twenty three bench mark functions. In [10], authors have proposed a new optimization algorithm i.e. whale optimization algorithm, WOA had been applied to twenty nine bench mark functions and six structural design problems and observed that WOA performs better as compared to conventional methods (PSO, GSA etc).

 In this paper, Among available EAs, applications of GA and GSAPSO have been used to optimize the rule base of FLS and obtained optimum solutions for time series prediction. Further, each proposed application have been compared with bench mark data sets.

The rest of the paper is organized as follows, section II GA and hybrid PSO is introduced. Section III briefly describes Mackey Glass Time series. Section IV discussed Fuzzy Logic System. In section V modelling of time series with two evolutionary algorithms have been discussed , implementation and results generated from proposed algorithms have been presented in section VI and Conclusion has been done in section VII.

2. EVOLUTIONARY ALGORITHMS

2.1. Genetic Algorithm

GA is a search-based algorithm based on the concepts of natural selection and reproduction to get an optimum solution. GA is a subset of stochastic computation, i.e., Evolutionary

Computation [11] [12]. Applications of GA have been proposed for large spectrum of problems with a high degree of success.

In GAs, a pool or a population with possible solutions are considered to be an input for any given problem. These solutions interact with each other in controlled manner and produce better solutions, the process is repeated over for number of iterations due to its stochastic behavior. Each individual (or candidate solution) is assigned with a fitness value (based on its objective function value) and the fittest individuals among all will be considered for higher chance to mate and yield more "fittest" individuals. Application of GA implemented for time series prediction in this paper is shown in Fig.1 and pseudo code is given in Fig.2
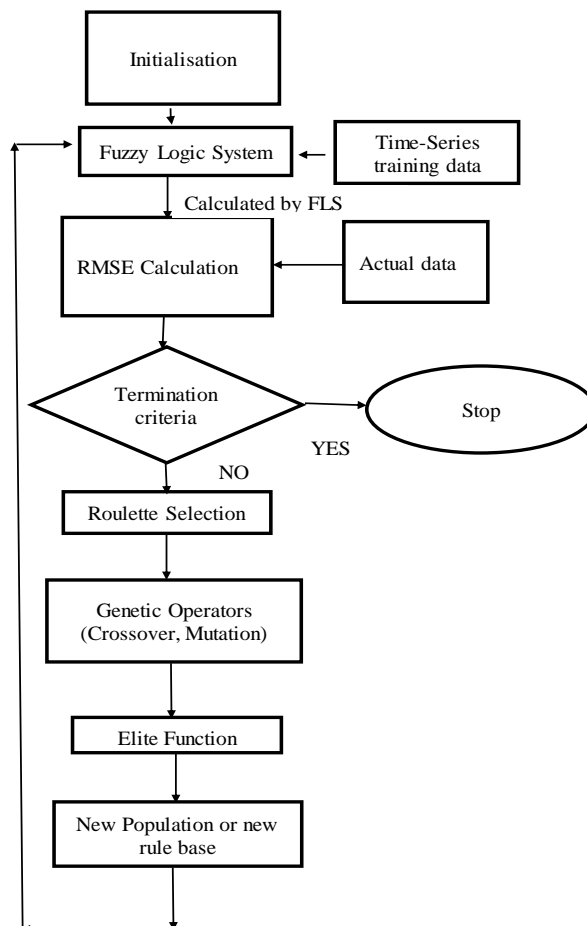

Figure 1 Flowchart for FLS GA

The following steps have been followed for the proposed algorithm:

1. Initialize uniform random population.
2. Select population using Roulette Wheel selection method.
3. Perform uniform crossover.
4. Perform mutation.
5. Find error i.e. RMSE using FLS.
6. Repeat for N iterations.

The pseudo code for GA is given below:

GA-FLS Pseudo Code

Initialize the population X_i (i = 1, 2,…, n)

Evaluate fitness of individual by using

calculate_rmse() as objective function.

Set mean and sigma matrices.

Iteration t=1

While (t<max_no_iterations)

Select the individual according to calculate_rmse() function

Perform uniform crossover on the population

Perform mutation on the population

Perform Elite Function

Population = new population

t=t+1

end while

Figure 2 FLS and GA Pseudo

In this way we keep "evolving" better individuals or solutions over generations, till we reach a stopping criterion.

2.2. Hybrid Particle Swarm Optimization

PSO is a stochastic global optimization technique that is based on the social behavior of bird flocking or fish schooling. PSO was proposed by Eberhart and Kennedy in 1995 [13]. The fundamental concept behind PSO is that each particle represents a potential solution which it updates according to two important kinds of information available in decision process. In this the particle will gain knowledge by its own experience and also gain experience of its neighbors. Particle tries the choices itself and has the knowledge which choices its neighbors have outstand so far and how positive the best pattern of choices is adapted. PSO is being used increasingly due to its several advantages like robustness, efficiency and simplicity.

A novel approach with combination of PSO and Gravitational Search Algorithm (GSA) is proposed by [9] which increases strength to get an optimum solution. In this paper, proposed approach application has been implemented for the time series prediction. According to Talbi several hybridization methods are possible [14]. As per research two algorithms can be hybridized in high level or low-level with relay or co-evolutionary method as homogenous or heterogeneous. Such implementations use low-level heterogeneous hybrid. The basic idea is to take advantage of social thinking of PSO and local search capability of GSA. In \ref{sarab} C1,C2 are learning parameters, V(t) is the velocity of the particle in previous iteration V(t+1) is the velocity of particle after next iteration, rand is random number between [0,1], ac is the acceleration of the particle, X(t+1) is the final position of the particle and X(t) is the previous position of the particle. Pseudo code for the HPSO is shown in Fig. 4 and flow chart for HPSOGSA implementation for time series prediction is shown in Fig.3

In each iteration the velocity and position of particles are updated by the following equations

$$V_i(t+1) = V_i(t) + C_1 * rand * ac_i(t) + c_2 * rand * (gbest - X_i(t)) \qquad (1)$$

All agents use 1 to update their velocity. Then next position of any agent is calculates using the equation:

$$X_i(t+1) = V_i(t+1) + X_i(t) \qquad (2)$$

All agents are initialized randomly in PSOGSA [9]. Every agent is considered as candidate solution. After initializing various parameters Gravitational constant, gravitational force and resultant forces among agents are calculated. After calculating the forces acceleration of various agents are calculated. The difference between standard PSO and GSAPSO lies in the velocity calculation of agents. In standard PSO acceleration does not have any effect on agent velocity whereas in GSAPSO acceleration effect the velocity of agent computed using GSA. This will have impact on the position of the agents. The steps of GSAPSO is shown in Figure 3
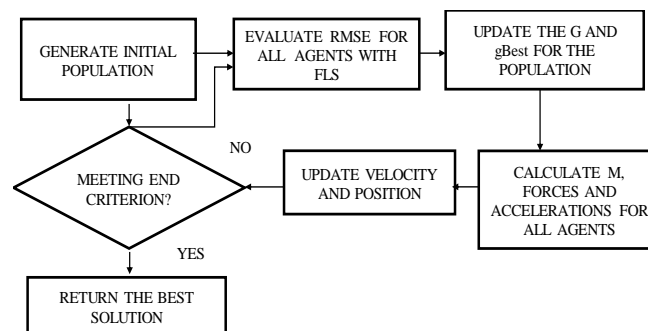

Figure 3 Flowchart for FLS and Hybrid PSOGSA

The pseudo code for HPSO is shown below

> **HPSO-FLS Pseudo Code**
>
> *Initialize the population X_i (i = 1, 2,..., n)*
>
> *t=1*
>
> *While (t<max_no_iterations)*
>
> *Evaluate fitness of all agents i.e. RMSE*
>
> *Update G and gbest for the population*
>
> *Calculate M, Forces and Acceleration for particles.*
>
> *Update Velocity and Position of search agents*
>
> *Update gbest for the solution*
>
> *t=t+1*
>
> *end while*

Figure 4 FLS and HPSO Pseudo

3. MACKEY GLASS TIME SERIES

Mackey Glass Time Series is a chaotic system described by the following equation:

$$x(t) = ax(t - \tau)/(1 + x^{10}(t - \tau) - bx(t)) \qquad (10)$$

Where τ, a, b are parameters of the system. The system is chaotic when τ=17, a=0.2, b=0.1.

Given the equation 10 the task is to predict the value of time series at point x(t+1) from earlier points x(t-18), x(t-12), x(t-6), x(t). This is expressed by the equation:

$$x(t) = f(x(t-18), x(t-12), x(t-6), x(t)) \qquad (11)$$

In order to train an FLS 700 data pairs were taken to evolve the rule base. Once rule base is evolved the it is used to predict the time series.

4. FUZZY LOGIC SYSTEM

Fuzzy logic is one of the most effective way to mimic human expertise in a naturalistic way [15]. Computing with words help us to develop models mathematically of the events that has been articulated using language. Fuzzy logic increases the robustness of the system in that it arrives at an accommodation with the imprecision pervasive in any real-world system. The basic configuration of the FLS consist of inputs and outputs scaling factors, rule base, inference and a defuzzifier as shown in the Fig. 5.

The structure of FLS rule base can be modeled as in 12.

$$R_i : if \ x_1 \ is \ A_{i1} \ and \ x_2 \ is \ A_{i2} \ and \ \dots \ x_n \ is \ A_{in} \ then \ y_i \ is \ W_i \qquad (12)$$

Where R_i is the ith rule, x_j is the jth input variable of FLS: n is the number of input variables, y_i is the ith output variable; A_{ij} are the fuzzy membership functions for inputs and W_i are the membership functions of out variables of sugeno type systems.

In this paper Sugeno type systems has been used as these are more accurate. Min aggregation and height defuzzifier method have been used [16][17]. For any input X=(x1, x2........) then y is the crisp, i.e., of the FLS 13 while the defuzzifier can be expressed in 14

$$\mu(y) = \min(\min(A_{in}, W_i)) \qquad (13)$$

$$y_h = \frac{\left[\sum_{l=1}^{M} y^{-1} \mu_B^{l(y^{-1})}\right]}{\left[\sum_{l=1}^{M} \mu_B^{l(y^{-1})}\right]} \qquad (14)$$
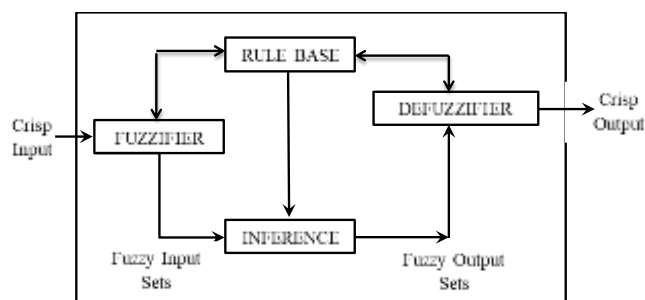

Figure 5 Fuzzy Logic System

The complete rule base of the system is shown in Fig 6. There are total four antecedents. Each antecedent has been modeled using three fuzzy sets making total of 81 different possible rules. Out of these 81 only 10 are selected for the rule base as each rule must be distinct. There are total ten consequents in the fuzzy rule base. after each iteration rule base is checked to contain distinct rules.

a11 (1-3)	a12(1-3)	a13(1-3)	a14(1-3)
a21 (1-3)	a22(1-3)	a23(1-3)	a24(1-3)
·			
·			
·			
·			
·			
a81,1(1-3)	a81,2(1-3)	a81,3(1-3)	a81,4(1-3)

c1 (1-10)
c2(1-10)
c3 (1-10)
c4(1-10)
c5 (1-10)
c61-10)
c7 (1-10)
c8(1-10)
c9(1-10)
c10(1-10)

Figure 6 Complete Rule base

5. MODELLING TIME SERIES FORECASTING USING FLS WITH APPLICATION OF GA AND GSAPSO

In this section, time series data set has been modeled to be linear with FLS system and further optimized using the application of GA and GSAPSO algorithms.

5.1. Fuzzy modelling for time series prediction with application of GA and GAPSO

In this paper Ten fuzzy rules from forty different rule bases have been evolved using standard data set. Each rule contains 4 antecedents and 1 consequent. Each antecedent is modelled using three fuzzy sets. The membership functions used for fuzzification of time series forecasting are Z-type, Gaussian type and S-type. If the value in the antecedent part of rule is 1 by min operator for rule composition, then Z-type membership function is selected and further its min operator is used for the input data. Similarly, if value of antecedent part is 2 after min operator composition then gaussian is selected and its min operator is generated which will be followed by S-shaped if the value is 3. The min operator has been used for the antecedent part rule composition and final value is considered for a rule (only for antecedent part). Ten rules have been evolved for a single consequent to map all the possibilities of an FLS and further height defuzzifier is used to get single output of time series model.

In this paper, 700 data sets have been used and correspondingly 700 error values have been generated. Further these error values are summed up to find mean square error (MSE) and then root mean square error (RMSE). The main objective of proposed FLS system is to find the predicted value for the data set with min RMSE value from the evolved rule base using evolutionary algorithm.

The sample of rule base used in the paper is shown in table 1.

Table 1 Sample rule base evolved

R1	a11	a12	a13	a14	c11	……….	a11	a12	a13	a14	c1n
R2	.										
	.										
.	.										
Rn	an1	an2	an3	an4	cn1		an1	an2	an3	an4	cnn

5.2. Rule base with GA and GA parameters

GA involves multiple solutions represented by a string of variables, analogous to the chromosomes in Genetics. Initially population is generated randomly, every swarm member gets a chance to evolve. Evolution of a chromosome is based on the fitness pairs of parent

solutions that has been selected randomly to reproduce next generation of solutions, stochastically. Each child chromosome has features of both the parent as an output of crossover. Another minor alteration in feature values of the generation represents effect of mutation.

The following parameters have been used for the proposed system:

1. Universe of discourse between 1-3 for antecedents and 1-10 for consequents.
2. Selection method Roulette Wheel selection.
3. Crossover parameter=0.80
4. Mutation parameter = 0.05.
5. Population size= 40
6. Number of chromosomes= 50.
7. Min threshold = 0.02
8. Number of iterations= 500
9. Input data points=700
10. Number of trials=30.

The rule base evolved using GA FLS is shown in the table 2.

Table 2 Rule base of GA

I1	I2	I3	I4	O
2	1	2	2	8
1	2	1	2	3
2	2	1	2	2
3	2	2	2	6
2	2	2	1	4
2	3	1	2	9
2	2	1	1	1
2	3	2	3	10
3	3	1	3	5
2	1	3	2	7

In table 2, I1, I2, I3, I4 are the value of antecedents and O is the consequent value of rule Ri. Each rule may contain a consequent value from 10 possible consequent values to compose the rule base for an individual input, i.e., string of 50 elements is randomly generated. In FLS system, total 40 rule bases are evolved at a time to compose the rule base for a time series prediction system. Further these 40 rule bases are optimized to minimize the RMSE values and MSE values.

5.3. Rule base with GSAPSO and parameters

In [9] two algorithms, PSO and GSA i.e. hybrid PSO had been proposed. The parameters used in [9] have been used in this paper for evolution of FLS rule base for time series prediction.

1. Universe of discourse between 1-3 for antecedents and 1-10 for consequents.
2. C1 =0.5, C2= 1.5. are learning factors.
3. Descending coefficient i.e. α =20
4. W uniform random number between [0,1].
5. Population size of 40

6. Starting value of Gravitational constant 1.
7. Number of iterations =500
8. Input data points=700.
9. Number of trials=30

The rule base evolved using HPSO is given in table 3.

Table 3 Rule base of PSOGSA

Best Rule Base of PSOGSA				
4	1	1	1	3
4	1	2	1	5
4	1	1	1	6
1	1	2	1	7
1	3	1	1	4
1	3	2	1	10
4	1	1	1	2
4	3	1	1	9
4	1	1	1	8
1	1	1	1	1

In the table 3, I1, I2, I3, I4 are the input values of rule base work as antecedents of FLS and O will provide the output vale of FLS for any arbitrary rule Ri. Rule base contains 10 different rules i.e. string of 50 each. Total 40 rule base are evolved at a time. Out of these only best one i.e. with minimum RMSE value is selected by the proposed system.

6. RESULTS AND DISCUSSION

The simulation of the proposed algorithm has been carried out with i5 -6200U CPU @ 2.30 GHZ, 8 GB random access memory, Windows 10 OS, Visual Studio 2010. In this paper application of recent evolutionary algorithms like GA and GSAPSO have been implemented to evolve the optimum fuzzy rule base for time series prediction. FCM has been used to design fuzzy membership functions from time series data set available for prediction to remove non-linearity from data set.

The simulation results of each proposed evolutionary algorithms have their individual performance with respect to RMSE and it is observed that among the two GA has performed better in terms of execution time and best RMSE value is attained.

The simulation result of GA with FLS are shown in Fig. 7. It is observed that the calculated output of the proposed algorithm is similar to the actual output as desired by the system. The mean error square value of GA FLS is 0.0001 and time of execution is 70 seconds 30 trials for 50k iterations, it is observed that time of execution and MSE value of GAFLS for prediction of time series forecasting system are acceptable. Further it is also observed that the proposed algorithm minimizes the computational complexity of a system and occupies the less memory due to the light behavior of nature.
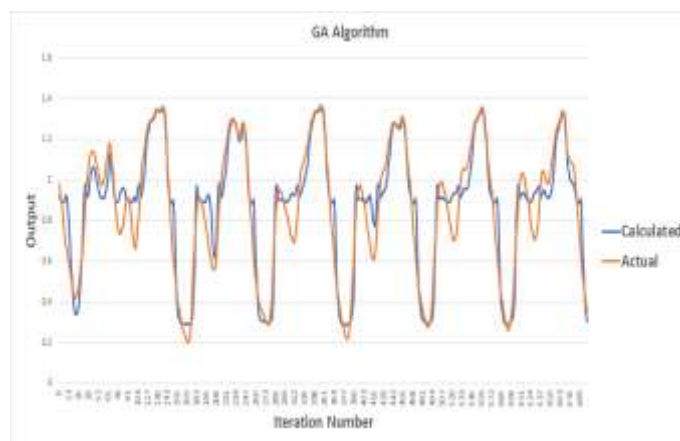
Figure 7 GA FLS output

Similarly, the simulation of GSAPSO with FLS have been carried out. In [9] GSAPSO, i.e., HPSO algorithm has been compared with PSO and other algorithms. The results of comparison show that HPSO outperforms standard PSO for different benchmark functions. The same HPSO with FLS has been simulated for time series forecasting. The simulations are carried out with different number of iterations for comparison of execution time. The mean square error is 0.0003 and execution is 90 seconds using 50k iterations 30 trials for GSA-PSO. It is observed that MSE value and execution time are acceptable for proposed application. The proposed algorithm converges fast and occupies less memory. The best output of HPSO is shown in the Fig. 8 from the figure it is concluded that calculated value for time series approximately matches the actual value of time series.
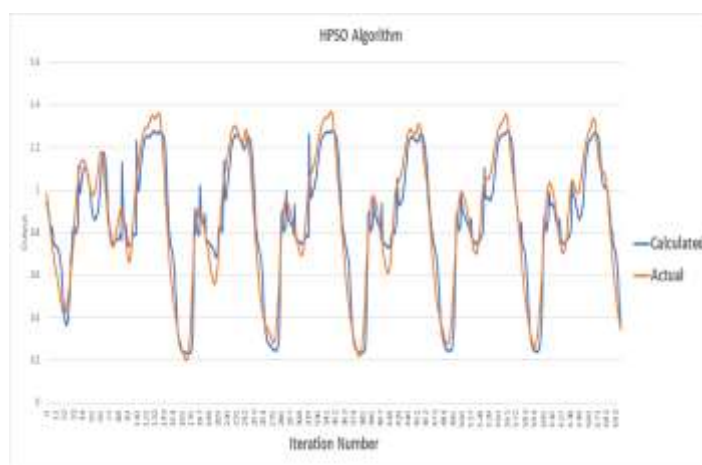

Figure 8 HPSO FLS output

7. CONCLUSION

Forecasting of Mackey Glass chaotic time series has been carried out in this paper. Further, application of two recent evolutionary techniques have been considered for optimization like Genetic Algorithm (GA), Particle Swarm Optimization (PSO) combined with GSA (PSOGSA), A comparison for bench mark data of time series forecasting is done using above discussed techniques. it is observed that GA performs better as compared to others in both terms, i.e. accuracy and time

REFERENCES

[1] M. C. Mackey, L. Glass, et al., Oscillation and chaos in physiological control systems, Science 197 (4300) (1977) 287–289.

[2] M. Bǖnner, T. Meyer, A. Kittel, J. Parisi, Recovery of the time-evolution equation of time-delay systems from time series, Physical Review E56 (5) (1997) 5083.

[3] L. Wang, F. Zou, X. Hei, D. Yang, D. Chen, Q. Jiang, Z. Cao, A hy- bridization of teaching–learning-based optimization and differential evo- lution for chaotic time series prediction, Neural computing and applica- tions 25 (6) (2014) 1407–1422.

[4] B. Samanta, Prediction of chaotic time series using computational intelli- gence, Expert Systems with Applications 38 (9) (2011) 11406–11411.

[5] W.-C. Juang, S.-J. Huang, F.-D. Huang, P.-W. Cheng, S.-R. Wann, Ap- plication of time series analysis in modelling and forecasting emergency department visits in a medical centre in southern taiwan, BMJ Open 7 (11) (2017) e018628.

[6] D. Karaboga, B. Basturk, A powerful and efficient algorithm for numer- ical function optimization: artificial bee colony (abc) algorithm, Journal of global optimization 39 (3) (2007) 459–471.

[7] I. Fister, I. Fister Jr, X.-S. Yang, J. Brest, A comprehensive review of firefly algorithms, Swarm and Evolutionary Computation 13 (2013) 34–46.

[8] S.-M. Chen, G. M. T. Manalu, J.-S. Pan, H.-C. Liu, Fuzzy forecasting based on two-factors second-order fuzzy-trend logical relationship groups and particle swarm optimization techniques, IEEE Transactions on Cy- bernetics 43 (3) (2013) 1102–1117.

[9] S. Mirjalili, S. Z. M. Hashim, A new hybrid psogsa algorithm for function optimization, in: Computer and information application (ICCIA), 2010 international conference on, IEEE, 2010, pp. 374–377.

[10] S. Mirjalili, A. Lewis, The whale optimization algorithm, Advances in Engineering Software 95 (2016) 51–67.

[11] D. E. Goldberg, Genetic algorithms in search, optimization, and machine learning, 1989, Reading: Addison-Wesley.

[12] J. H. Holland, Genetic algorithms and the optimal allocation of trials, SIAM Journal on Computing 2 (2) (1973) 88–105.

[13] R. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, in: Micro Machine and Human Science, 1995. MHS'95., Proceedings of the Sixth International Symposium on, IEEE, 1995, pp. 39–43.

[14] E.-G. Talbi, A taxonomy of hybrid metaheuristics, Journal of heuristics 8 (5) (2002) 541–564.

[15] L. A. Zadeh, Fuzzy logic computing with words, IEEE transactions on fuzzy systems 4 (2) (1996) 103–111.

[16] T. J. Ross, Fuzzy logic with engineering applications, John Wiley & Sons, 2009.

[17] J. M. Mendel, Fuzzy Logic Systems for Engineering: A Tutorial, Pro- ceedings of the IEEE 83 (3) (1995) 345–377.