Identification of a Nonlinear System by Determining of Fuzzy Rules

Hodjatollah Hamidi*
Department of Industrial Engineering, K. N. Toosi University of Technology, Tehran, Iran
h_hamidi@kntu.ac.ir
Atefeh Daraei
Department of Industrial Engineering, K. N. Toosi University of Technology, Tehran, Iran
adaraei@mail.kntu.ac.ir

Received: 22/Apr/2016 Revised: 29/Jul/2016 Accepted: 10/Aug/2016

Abstract

In this article the hybrid optimization algorithm of differential evolution and particle swarm is introduced for designing the fuzzy rule base of a fuzzy controller. For a specific number of rules, a hybrid algorithm for optimizing all open parameters was used to reach maximum accuracy in training. The considered hybrid computational approach includes: opposition-based differential evolution algorithm and particle swarm optimization algorithm. To train a fuzzy system hich is employed for identification of a nonlinear system, the results show that the proposed hybrid algorithm approach demonstrates a better identification accuracy compared to other educational approaches in identification of the nonlinear system model. The example used in this article is the Mackey-Glass Chaotic System on which the proposed method is finally applied.

Keywords: System Identification; Combined Training; Fuzzy Rules; Database Design.

1. Introduction

System identification is extensively used in a number of programs including control systems [1], communications [2], signal processing [3], controlling chemical processes [4], biological processes [5] and etc. to be exact all problems of the real world are nonlinear per se. Anyhow, we face less computational problems in identifying a linear system and normally we are not faced with simple problems in identifying nonlinear systems. In [6] particle swarm optimization with different particle length is proposed for production of structure and parameters of fuzzy rule database. In [7], the continuous version of ant colony optimization is used for designing of fuzzy rules database. In this work the online method is used for determining the number of fuzzy rules and all open parameters in each fuzzy rule in the continuous space was continually optimized by ant colony optimization algorithm. In [8], system training is presented using two-step swarm intelligence algorithm. This algorithm includes two steps. In the first part structure and initial parameter identification is carried out using online clustering of ant colony optimization algorithm in disrupted space. In the second part particle swarm optimization is used for greater optimization of all open parameters in the continuous space.

Fuzzy systems are suitable for complex systems' modeling, due to the good feature of general approximation especially for systems in which mathematical description is difficult. It is proven that any continuous function can approximate a logical degree of accuracy using a fuzzy system which is trained by meta-heuristic algorithms. This approximation function can act as a model for a number of functional complex systems. Juang et al, 2014, have shown that fuzzy systems that are trained by algorithm can be effectively used in identification of nonlinear models.

Recently, the differential evolution algorithm (DE) is considered as a modern technique of evolution calculations [9,10] that are used for optimization issues. DE is preferred over other evolution methods like genetic algorithm (GA) [11,12] and particle swarm optimization (PSO) and this is due to its notable characteristics like simple concept, easy execution and rapid convergence [13,14]. Generally all population-based optimization algorithms which also include DE suffer the long calculation period due to their evolutionary-accidental nature.

The concept of opposition-based learning (OBL) is introduced by Tizhoosh [15]. In this article, this concept is used for acceleration of learning in fuzzy systems. The main idea in OBL concept is simultaneous consideration of an estimate and its corresponding opposing estimate. OBL leads to achievement of a better estimation and acceleration the rate of DE convergences. PSO is an algorithm with local search pattern and can be used to fine-tune the present results and faster access to global minimum. Therefore the proposed method in this article is called hybrid opposition-based differential evolution with particle swarm optimization (HODEPSO). ODE utilizes opposing numbers during the start of population and also for production of new population during the evolution process. Here, opposing numbers are used to accelerate the rate of convergences of DE optimizing algorithm. Pure random sampling or selection of solutions from data

* Corresponding Author

population provides for a chance to visit or even inspection of undiscovered regions of the search space. It has been proven that the probability of this incident is less for opposing numbers than purely random numbers. In fact mathematical proof has been used to show that the probability of opposing numbers being closer to desired solutions is higher than completely pure numbers [16-19]. In [17], the benefit of opposing numbers is investigated by replacing them with random numbers and this method has been utilized for initializing the population and generation skipping for different DE versions.

This article presents a new educational sample of fuzzy systems which are combined with meta-heuristic evolutionary algorithm, meaning:

- Use of Opposition-Based Learning concept as simultaneous considering of an estimate and its opposing corresponding estimate which would lead to better estimation and acceleration of the rate of convergence of differential evolution algorithm (DE).
- Combination of ODE and PSO to prevent the probability of getting caught up in local optimum and quicker and more accurate achievement of general optimum.
- Performance of the trained fuzzy system using HODEPSO is shown by comparing the results of some of the present methods in the un-linear system identification. Results of stimulation, shows the suitable performance of the proposed method compared to other methods of identification.

2. Meta- Heuristic Optimization (MHO) Algorithms

Optimization methods are search methods that aim at finding answers to the optimization problem so that the evaluated quantity is optimized. According to evidence and records of results, the best quality and time opposition for fuzzy system optimization is provided using meta-heuristic algorithms.

2.1 Differential Evolution (DE) Algorithm

Differential evolution algorithm (DE) is one of the effective search-based methods [20-33]. Like other evolution algorithms, this one also starts by initializing a population. Then through implementation of agents like combination, mutation and generation convergence, the new-born is formed and in the next step which is called selection, new born generation is compared to parent generation to determine the rate of aptitude which is evaluated by the goal function. Then the best members enter the next round as the next generation. This trend continues until desired results are reached. Different levels of this algorithm are stated here in sequence.

Population Initialization: The number of variables in this algorithm are shown with D. each of these variables hold a high and low limit. Initial population with the size of N_P in D is randomly formed according to equation (1).

$$X_{io} = X_i \min + round\,(\delta_i.(X_i \max - X_i \min)) \quad ,i = 1,2,\ldots,N_P \tag{1}$$

Where δ_i is a random number in the (0,1] domain, X_i max and X_i min are the high and low limits of the variables and N_P is the number of members.

Mutation and Intersection: in this algorithm five strategies can be utilized for combination and production of new-borns [18]. In this article best person-random person- random person is used for mutation as follows:

$$Z_{i,G} = X_{best,G} + F.(X_{r1,G} - X_{r2,G} + X_{r3,G} - X_{r4,G}) \tag{2}$$

Where F is called standard factor, X_2s are randomly selected members and Xbest is the best member of the present population.

For every variable of each member of the population a random number, K, in the [1, D] domain and a random number, u, in the [0, 1] domain is selected. Intersection is carried out according to the equation below:

$$\begin{aligned} &if \quad u \le CR \quad or \quad j = k \\ &then \quad Z_{i,j} = X_{r1,j} + F(X_{r3,j} - X_{r2,j}) \\ &else \quad Z_{i,j} = X_{i,j} \end{aligned} \tag{3}$$

Where j is the number of any variable from i^{th} member of the population and CR is the intersection constant and is chosen as a number between 0 and 1.

Estimation and selection: at this stage the new-borns and parents are valuated according to the goal function and if the newborn has a higher value than the parent, it replaces the parent. Otherwise the parent moves on to the next level with the next generation.

$$Z_{i,g+1} = \arg\max(\, f(z_{i,}g), z_{i,}g+1) \tag{4}$$

In this equation g stands for generation, $Z_{i,g+1}$ is the new generation population (new-borns) and $Z_{i,g}$ is the previous generation population (parents). F is the goal function of the problem.

Repetition: repeating steps 2 and 3 until maximum repetition or the whole population convergence is reached.

2.2 Opposition-Based Differentiation Evolution (ODE) Algorithm

In optimization approaches of evolution algorithm, a unified random guess for the initial population is considered. In each generation the goal includes movement towards the desired solution and the research trend ends when some of the pre-determined criterion are satisfactory. Calculation time usually depends on initial guess, meaning that the greater the distance between initial guess and desired solution, the more time it takes to reach the end and vice versa. Opposition-based learning increases the chance to start with a better initial population through revision of opposing solutions.

Similar approaches to this can be used not only in initial solutions but also utilized continually in the present population for any solution [19].

2.2.1 Definition of Opposing Number

Suppose $x \in [a, b]$ is a real number. The opposing number is $\tilde{x}$ which is defined by $\tilde{x} = a + b - x$.

Definition of opposing point: suppose $p = (x_1, x_2, \ldots, x_d)$ is a point in a D-dimensional space in which $x_1, x_2, \ldots, x_d \in R$ and $x_i \in [a_i, b_i]$. Opposing point is:

$$\tilde{p} = (\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_d) \text{ where } \tilde{x}_i = a_i + b_i - x_i$$

2.2.2 Opposition-Based Optimization (OBO)

Suppose $p = (x_1, x_2, \ldots, x_d)$ is a point in a d-dimensional space, meaning suppose an elective solution. F(0) is a proportion function which is used to measure the proportion of selections. According to definition of opposing points, $\tilde{p} = (\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_d)$ opposes $p = (x_1, x_2, \ldots, x_d)$. Now, if $(\tilde{p}) \geq f(p)$, then p can be replaced by $\tilde{p}$ otherwise we continue with p. therefore the point is evaluated simultaneously with its opposing point so that we continue the algorithm with the more suitable ones.

2.3 Particle Swarm Optimization (PSO) Algorithm

Particle Swarm Optimization Algorithm (PSO) works according to the social behaviour of birds [20]. For better understanding of this technique, consider the below scenario: "a flock of birds are randomly looking for food in a specific region. There is only one piece of food in this region which the birds are not aware of but are aware of their distance with the food all the time." At this state, a suitable strategy for fining the exact location of food is following the bird that is closer to the food. Actually PSO has been inspired by such a scenario too and presents a solution for optimization problems in PSO each bird is a solution to the problem. All the present responses have a fitness value which is calculated by the defined fitness function for the problem. The aim of this technique is finding a location with the best fitness value in the problem setting. This fitness value has a direct effect on the direction and speed of these birds' movement (solutions to the problem) towards the location of the food (optimal response).

PSO starts with a number of initial response (particles) and looks for optimal response by moving these responses in continuous repetitions. In every repetition two values are determined: P_{Best} and G_{Best}.

P_{Best}: Location of the best P_{Best} fitness value where each [article has reached in its movement,

G_{Best}: Location of the best particle fitness in the present population.

After the above values are calculated, the particles' speed of movement is calculated by equation (4) and each particle's next location is calculated by equation (5).

$$\begin{aligned} \mathrm{V}_{\mathrm{i,t+1}} &= w.v_{i,t-1} + c_1.r_1.(P_{best_i} - P_{i,t}) \\ &+ c_2.r_2.(G_{best_i} - P_{i,t}) \end{aligned} \quad (5)$$

$$\mathrm{P}_{\mathrm{t+1}} = P_t + v_t \quad (6)$$

In these equations r_1 and r_2 values are random numbers between zero and one and c_1 and c_2 coefficients which are called learning coefficients are usually equalled to two initializations.

In every repetition of algorithm, the speed of particle movement (rate of change for each particle) in every dimension can be limited with a pre-determined V_{max} value. At this state if the speed of each particle in each dimension exceeds this limit, we replace it by V_{max}.

3. Hybrid Opposition-Based Differential Evolution and Particle Swarm Optimization Algorithms

In this article, the hybrid algorithm of opposition-based differential evolution and particle swarm optimization (HODEPSO) is effectively developed. The exact details of steps in hybrid algorithm of opposition-based differential evolution and particle swarm optimization is explained below:

Step 1: Random population initialization and by considering simultaneous Gaussian of opposing initial values.

Step 2: enforcement of opposition-based differential evolution algorithm agents on the initial population.

Step 3: Evaluation of the cost function (which is as RMSE in the solved example in this article) for each particle and updating P_{Best} and G_{Best}.

Step 4: Selection of parents' and Gaussian their opposition and enforcement of opposition-based differential evolution algorithm agents on them.

Step 5: Evaluation of cost function for the new-borns and updating P_{Best} and G_{Best} particle speed.

Step 6: After selection of new-borns from the elected parents, the survivor selection mechanism is performed.

Step 7: updating particle speed and status using equations (5) and (6).

Step 8: Evaluation of cost function for each particle and updating P_{Best} and G_{Best}.

Step 9: If the conditions for ending is established, hybrid algorithm can end otherwise go to step 4.

4. Fuzzy System

In this section, the design of the hybrid algorithm-based fuzzy system is described. The fuzzy system used if of TSK, zero-degree kind in which the i[th] fuzzy rule is specified with R_i and described as follows:

$$R_i \; : \; If \; x_1 \; is \, A_1^i(x_1) \; and \; x_2 \; is \, A_2^i(x_2) \, then \, y \; is \; B^i \quad (7)$$

Where R_i is the i^{th} fuzzy rule, x_j is the input variable, y the output variable, $A_{ij}(x_j)$ is the fuzzy set and B is a certain value. The fuzzy set $A_{ij}(x_j)$ is a Gaussian membership function described by the equation below:

$$A_{ij}(x_j) = exp\left\{-\left(\frac{x_j - m_{ij}}{\sigma_{ij}}\right)^2\right\} \tag{8}$$

Where m is the center and is the width of the Gaussian membership function. For x_1 and x_2 inputs, the meridian or effective weight w_i is calculated as follows:

$$w_i = A_1^i(x_{1k}) + A_2^i(x_{2k}) \tag{9}$$

If the fuzzy system has r rules, the fizzy system output is calculated with the defuzzication weighted average as follows:

$$y_i = \frac{\sum_{i=1}^{r} w_i . B^i}{\sum_{i=1}^{r} w_i} \tag{10}$$

The status of each particle is stated with the vector below in the search space:

$$P = [\, m_{11}, \sigma_{11}, m_{12}, \sigma_{12}, a_1, m_{21}, \sigma_{21}, m_{22}, \sigma_{22}, a_2 \,] \tag{11}$$

Where a is the tally value in each fuzzy law. For example if the fuzzy system has n input variables and r is the rule, the number of vector member, p (number of optimizing variables) would equal $(2n + 1) \times r$.

5. Results of Simulation

In this example the designed fuzzy system is used to predict future values of Mackey-Glass chaotic time series. This time series is produced using the Mackey-Glass delay differential equation as below:

$$\dot{x}(t) = \frac{0.2x(t-\tau)}{1 + x^{10}(t-\tau)} - 0.1x(t) \tag{12}$$

The issue of predicting time series based on Mackey-Glass differential equation is a famous criterion for comparison of capacities of different fuzzy models. 1000 pairs of input-output data were extracted from Mackey-Glass chaotic time series. The first 500 pairs were used for trainienging of the fuzzy system while the remaining 500 were used as test data to evaluate the performance of the fuzzy model in prediction.

$$\begin{aligned} &Input\ 1 = x(t-30) \\ &Input\ 2 = x(t-18) \\ &Input\ 3 = x(t-12) \\ &Input\ 4 = x(t) \end{aligned} \tag{13}$$

To evaluate the performance of the designed fuzzy model RMSE was used which is defined as follows:

$$RMSE = \sqrt{\frac{1}{n}\sum_{k=1}^{n}(y_k - \hat{y}_k)^2} \tag{14}$$

Where n is the number of data, y_k is the real output and $\tilde{y}_k$ is the fuzzy model output.

To show the efficiency of hybrid algorithm compared to algorithms of opposition-based differential evolution and particle swarm optimization, each algorithm was used individually. The results of the use of these algorithms are shown in table (1) with an average of 50 times use.

According to the results, the designed fuzzy system with the proposed method, in addition to simplicity (reduction in the number of fuzzy rules) shows a better performance compared to fuzzy models presented in [21] and [22] and has achieved a lower RMSE compared to those.

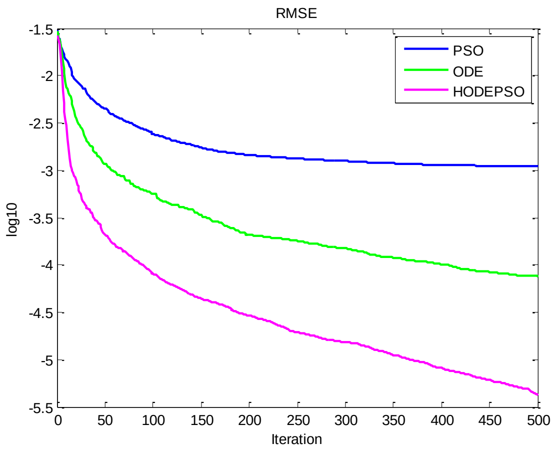

Fig. 1. RMSE values achieved in every repetition by PSO, ODE and HODEPSO.

According to Fig.1 it is observed that the hybrid algorithm converges more quickly to the optimal solution and has better performance compared to individual ODE and PSO algorithms.

The same example is studied in references [21] and [22]. Comparison of the results of these methods and the proposed method are shown in table (1).

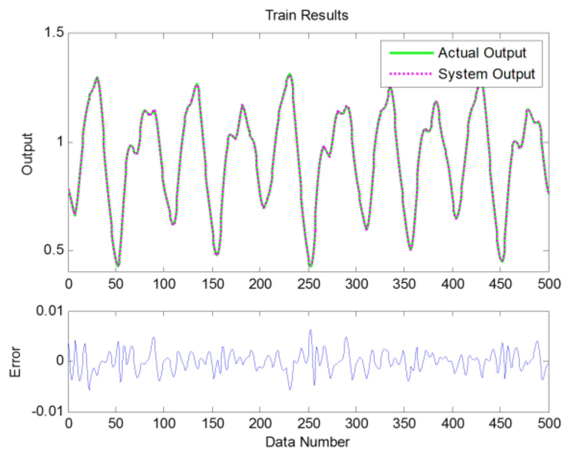

Fig. 2. Comparison of real output and fuzzy model output for train data

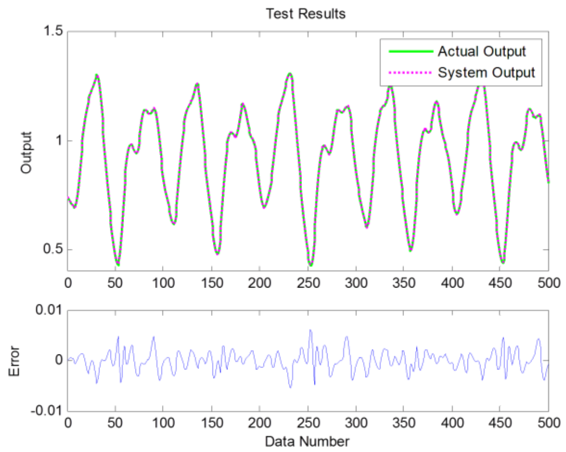

Fig. 3. comparison of real output and fuzzy model output for test data

Table 1: Comparison of results of different methods

Method	No. of rules	RMSE trainig	RMSE Test
PSO	10	1.100E-3	2.300E-3
ODE	10	7.580E-5	9.340E-5
HODEPO	10	4.140E-6	8.360E-6
[21]	4	0.0094	0.0061
[21]	10	-	0.0039
[22]	3	0.00588	0.00587

6. Conclusion

In this article the hybrid optimization algorithm of differential evolution and particle swarm is introduced for designing the fuzzy rule base of a fuzzy controller. For a specific number of rules, a hybrid algorithm for optimizing all open parameters was used to reach maximum accuracy in training. The aim of using this algorithm was to set the parameters of the rule base in the zero-degree fuzzy system (TSK) in order to minimize the performance index (Root Mean Square Error (RMSE)).using the mentioned algorithm, the time-consuming process of parameter adjustment became a simple and quick task. The results show the suitable performance of the proposed model compared to other methods.

References

[1] S. Chen, S.A. Billings, Representation of non-linear systems: the NARMAX model, Int. J. Control 49 (1989) 1013–1032.

[2] H. Hujiberts, H. Nijmeijer, R. Willems, System identification in communication with chaotic systems, IEEE Trans. Circuits Syst. I 47 (6), 2000, pp. 800–808.

[3] M. Adjrad, A. Belouchrani, Estimation of multi component polynomial phase signals impinging on a multisensor array using state-space modeling, IEEE Trans. Signal Process. 55 (1), 2007.pp. 32–45.

[4] K. Watanbe, I. Matsuura, M. Abe, M. Kebota, D.M. Himelblau, Incipient fault diagnosis of chemical processes via artificial neural networks, AICHE J. 35 (11) (1989) 1803–1812.

[5] Y. Xie, B. Guo, L. Xu, J. Li, P. Stoica, Multistatic adaptive microwave imaging for early breast cancer detection, IEEE Trans. Biomed. Eng. 53 (8), 2006, pp. 1647–1657.

[6] D. Chen, J. Wang, F. Zou, H. Zhang, W. Hou, “Linguistic fuzzy model identification based on PSO with different length of particles”, Applied Soft Computing 12, pp. 3390–3400, 2012.

[7] C. F. Juang, and P. H. Chang, “Designing Fuzzy-Rule-Based Systems Using Continuous Ant-Colony Optimization”, IEEE Trans on Fuzzy Syst., vol. 18, no. 1, pp. 138-149, Feb 2010.

[8] C. F. Juang, C. Lo, “Zero-order TSK-type fuzzy system learning using a two-phase swarm intelligence algorithm”, Fuzzy Sets and Systems 159, 2910 – 2926, 2008.

[9] R. Storn, System design by constraint adaptation and differential evolution, IEEE Trans. Evol. Comput. 3 (1999) 22–34.

[10] J. Ilonen, J.K. Kamarainen, J. Lampinen, Differential evolution training algorithm for feed forward neural networks, Neural Proc. Lett. 17 (2003) 93–105.

[11] E. Goldberg, J. Richardson, Genetic algorithms with sharing for multimodal function optimization, in: J. Richardson (Ed.), Genetic Algorithms and their Applications (ICGA’87)., 1987, pp. 41–49.

[12] K. Kristinsson, G.A. Dumont, System identification and control using genetic algorithms, IEEE Trans. Syst. Man Cybernet. 22 (1992) 1033–1046.

[13] J. Ilonen, J.K. Kamarainen, J. Lampinen, Differential evolution training algorithm for feed forward neural networks, Neural Proc. Lett. 17, 2003, pp.93–105.

[14] R. Storn, System design by constraint adaptation and differential evolution, IEEE Trans. Evol. Comput. 3, 1999, pp. 22–34.

[15] H.R. Tizhoosh, Opposition-based learning: a new scheme for machine intelligence, in: Proc. Int. Conf. Comput. Intell. Modeling Control and Autom, vol. I, Vienna, Austria, 2005, pp. 695–701.

[16] S. Rahnamayan, H.R. Tizhoosh, M.M.A. Salama, Opposition Versus Randomness in Soft Computing Techniques, Elsevier J. Appl. Soft Comput. 8 (March (2)), 2008, pp. 906–918.

[17] S. Rahnamayan, H.R. Tizhoosh, M.M.A. Salama, Opposition-based differential evolution, IEEE Trans. Evol. Comput. 12 (1), 2008.

[18] K. V. Price, R. M. Storn, and J. A. Lampinen, "Differential Evolution: A Practical Approach to Global Optimization", (Kindle Edition). Springer, 2005.

[19] H.R. Tizhoosh, Opposition-based learning: a new scheme for machine intelligence, in: Proc. Int. Conf. Comput. Intell. Modeling Control and Autom, vol. I, Vienna, Austria, 2005, pp. 695–701.

[20] J. Kennedy, R. Eberhart, “Particle swarm optimization”, in: Proc. IEEE Internat. Conf. Neural Networks, Perth, Australia, pp. 1942–1948, 1995.

[21] C. F. Juang, C. W. Hung and C. H. Hsu, “Rule-Based Cooperative Continuous Ant Colony Optimization to Improve the Accuracy of Fuzzy System Design”, IEEE TRANSACTIONS ON FUZZY SYSTEMS, VOL. 22, NO. 4, AUGUST 2014

[22] W. Zhao, Q. Niu, K. Li and G. W. Irwin, “A Hybrid Learning Method for Constructing Compact Rule-Based Fuzzy Models”, IEEE TRANSACTIONS ON CYBERNETICS, VOL. 43, NO. 6, DECEMBER 2013.

[23] H.Hamidi, “A Model for Impact of Organizational Project Benefits Management and its Impact on End User”, JOEUC, Volume 29, Issue 1, 2017, pp.50-64.

[24] K.Mohammadi, H.Hamidi., Modeling and Evolution of Fault-Tolerant Mobile Agents in Distributed System.The Second IEEE and IFIP International Conference on wireless and Optical Communications Networks (WOCN 2005), March 6 –8, 2005.

[25] S. A.Monadjemi, H.Hamidi, A.Vafaei. Analysis and Evaluation of a New Algorithm Based Fault Tolerance for Computing Systems. International Journal of Grid and High Performance Computing (IJGHPC), 4(1), 2012, pp. 37-51.

[26] S. A.Monadjemi, H..Hamidi., A.Vafaei. “ANALYSIS AND DESIGN OF AN ABFT AND PARITY-CHECKING TECHNIQUE IN HIGH PERFORMANCE COMPUTING SYSTEMS” Journal of Circuits, Systems, and Computers (JCSC), JCSC Volume 21 Number 3, 2012.

[27] A.Vafaei., S. A.Monadjemi, H..Hamidi., Evaluation of Fault Tolerant Mobile Agents in Distributed Systems. International Journal of Intelligent Information Technologies (IJIIT), 5(1), 2009, pp.43-60.

[28] A.Vafaei, S. A.Monadjemi, H.Hamidi “Evaluation and Check pointing of Fault Tolerant Mobile Agents Execution in Distributed Systems,” Journal of Networks, VOL. 5, NO. 7. 2009.

[29] H.Hamidi, A New Method for Transformation Techniques in Secure Information Systems Journal of Information Systems and Telecommunication, Vol. 4, No. 1, January-March 2016, pp. 19-26.

[30] X. Ye, T.Sakurai, “Robust Similarity Measure for Spectral Clustering Based on Shared Neighbors,” ETRI Journal, vol. 38, no. 3, June. 2016, pp. 540-550.

[31] J.Wu, F. Ding, M. Xu, Z. Mo, A..Jin. Investigating the Determinants of Decision-Making on Adoption of Public Cloud Computing in E-government. JGIM, 24(3), 2016, pp. 71-89.

[32] S. KUMAR, "Performance Evaluation of Novel AMDF-Based Pitch Detection Scheme," ETRI Journal, vol. 38, no. 3, June. 2016, pp. 425-434.

[33] B. Shadloo, A. Motevalian, V. Rahimi-movaghar, M.A. Esmaeili, V. Sharifi, A. Hajebi, R. Radgoodarzi, M. Hefazi, A. Rahimi- Movaghar, Psychiatric Disorders Are Associated with an Increased Risk of Injuries: Data from the Iranian Mental Health Survey, Iranian Journal of Public Health 45(5), 2016, pp. 623-635.

Hodjatollah Hamidi born 1978, in shazand Arak, Iran, He got his Ph.D in Computer Engineering. His main research interest areas are Information Technology, Fault-Tolerant systems (fault-tolerant computing, error control in digital designs) and applications and reliable and secure distributed systems, Machine learning, Knowledge Discovery and Data Mining. Since 2013 he has been a faculty member at the IT group of K. N. Toosi University of Technology, Tehran Iran. Information Technology Engineering Group, Department of Industrial Engineering, K. N.Toosi University of Technology.

Atefeh Daraei born in Khorram Abad, Lorestan Iran. She received her B.Sc in Information Technology in 2011 from University College of Nabi Akram, Tabriz, Iran. She is currently an M.Sc student in Information Technology (E-Commerce), at K. N. Toosi University of Technology, Tehran Iran. Her research interests include Machine learning, Knowledge Discovery and Data Mining and Customer Relationship Management.