Optimizing Risk Management Using Learning Automata

Babak Anari[1],Mohammad Reza Ahmadi[2], Mostafa gobaei Arani[3] and Zohreh Anari[4]

**[1] Computer Engineering, Shabestar branch, Islamic Azad University
Shabestar,Iran**

**[2] IT department, Research Institute of ICT (CSRI)
Tehran, Iran**

**[3] Computer Engineering, Kashan branch, Islamic Azad University
Kashan,Iran**

**[4] Computer Engineering, Shabestar branch,payame Noor University
Shabestar,Iran**

Abstract

Risk management processes are responsible for identifying, analyzing and evaluating risky scenarios and whether they should undergo control in order to satisfy a previously defined risk criterion. Risk specialists have to consider, at the same time, many operational aspects (decision variables) and objectives to decide which and when risk treatment have to be executed. Our objective is to automatically find a subset of risks that maximize risk reduction and respect the company operational resource limitations. This paper applied a Learning Automaton (LA) for risk reduction in uncertainly. To test the resulted methodology, experiments based on the Simple selection algorithm were performed aiming to manage risk and resources of a simulated company. Result show us that the proposed approach can deal with multiple conflicting objectives reducing the risk exposure time by selecting risks to be treated according their impact, and available resources.

Keywords: *Learning Automaton, Risk management, Risk optimization*

1. Introduction

Anything tangible or intangible that is capable of being owned or controlled to produce value and that is held to have positive economic value is considered an asset. Assets are susceptible to damages abused by undesired events. In line with that, risk management is the preventive process responsible for protecting companies against effects of these undesired events. Moreover these effects are what specialists call “risk”. ". Risk management processes identify, analyze, evaluate and treat risks aiming to protect assets. For each identified risk, an impact level, likelihood and treatment (or a set of treatments) are specified. Risk evaluation and treatment selection are processes that involve many strategic objectives according to the previously defined risk criteria, like: protection (reduction of the risk exposure time) or cost reduction. The process to decide the best approach to treat risks, considering multiple objectives and limited operational resources (e.g. time, money, human resources), is a complex task. Risk management process described by ISOs 31000 [1] and 31010 [2] specifies a general risk management process. These standards were built over an ideal scenario where a company has enough resources for treating risks over a period of time. Unlikely this ideal scenario, real world companies have limited operational resources. According to the amount of detected threats and resources, the treatment process has to be iterated over several time slices. A new issue rises from this scenario: risk selection optimization. A risk manager has to determine a set of risks to be treated which better satisfies all objectives and resource restrictions of the company. Aiming to help asset managers and companies to protect their business, this work investigates the use of a state-of-art multi-objective Learning Automaton for selecting treatments.

Now a days in [3], [4], [5], [8], [9], [10], [11], [12] and [13] algorithms for solving multi-objective combinatorial problems are proposed. The purpose of risk identification is finding, recognizing and recording risks. Risk analysis consists in determining the consequences and probabilities related to identify risks. The consequences and their probabilities are then combined to determine a level of the identified risk. The last step of risk assessment involves comparing analyzed risks with the risk criteria, in order to determine the significance and type of each risk. Risk assessment receives a scope as input and returns a list of evaluated risks. Finally, risk treatment step involves

selecting and agreeing to one or more relevant options for changing the probability of occurrence, the effect of risk, or both, and implementing these options. Risk treatment involves a cyclical process of:

- assessing risk treatment;

- deciding whether residual risk levels are tolerable; if not tolerable. Generating a new risk treatment; and assessing the effectiveness of this treatment. Aiming at identify new risks. Currently the entire ISO risk management process does not specify any order or priority for risk treatments execution. ISO risk management methodology defines steps for: (i) defining a scope (internal and external contexts), (ii) assessing these steps can be graphically visualized through Figure 1.

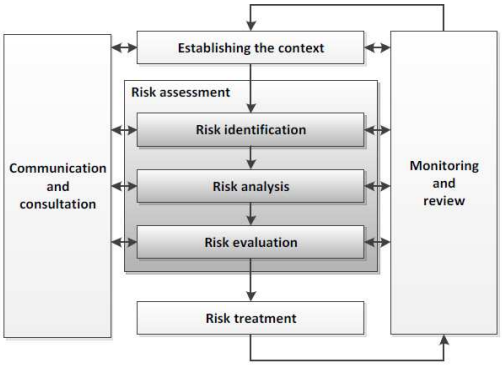

Figure 1: Risk Management Process

2. Information Security

Nowadays, some of the most valuable companies have information as their principal asset. Like any other kind of asset, information must be managed and can have associated risks, which need to be treated [7], [8]. Information Security domain of Security Management incorporates the identification of the information data assets with the development and implementation of policies, standards, guidelines and procedures. It addresses confidentiality, integrity and availability, by identifying threats, classifying the assets, and rating their vulnerabilities so that effective security controls can be implemented. The process described in Section II can be used for identifying, analyzing and evaluating risks associated to information sources. For each identified risk a security police is created (e.g. treatment). The concept of vulnerability was introduced and constitutes the absence of a safeguard. A minor threat has the potential to become greater threat, or a more frequent, because of vulnerability. Combined with the terms asset and threat, vulnerability is the third part of an element that is called a triple in risk management of information. Information security risks can be assessed by using several techniques based on historical data or specialist's opinion [2]. These identified risks can be treated by mitigating vulnerabilities and implementing security policies. Similarly to the general risk management process this specialized process does not specifies priorities and order for treating threats. Risk managers have to decide by themselves the best way to conduct the treatment process for balancing risk treatment and limited operational resources. No need to stress this is a fault prone process as it relies solely on the human abilities to tackle threats, which could be too many. According to the amount of detected threats and resources, the treatment process has to be iterated over several time slices. A new issue rises from this scenario: risk selection optimization. A risk manager has to determine a set of risks to be treated which better satisfies all objectives and resource restrictions of the company.

The proposed approach has three main objectives: (i) select treatments that best fit in an execution time window, (ii) better use financial resources and (iii) minimize risk exposure. For testing purposes, a simulated environment was developed containing assets, risks and treatments. This simulated environment returns a list of evaluated risks for an external optimization module deciding which risk will be treated. Although, experiments performed in this paper select risks according few objectives (enough for most companies), the proposed approach can be extended for optimizing risk selection considering several cooperative or competitive objectives. Preliminary results show that the proposed approach can successfully optimize the risk management process, reducing the risk exposure time by better using of operational resources. This article is divided into four main sections. In section 3 the learning automaton will be explained. The proposed architecture is presented in Section 4. In section 5 the effectiveness of the method is demonstrated. Finally we conclude in section 6.

3. Learning Automata

An automaton can be regarded as an abstract model that has finite number of actions. This action is applied to the selected action of automata. The random environment evaluates the applied action and gives a grade to the selected action of automata. The response from environment (i.e. grade of action) is used by automata to select its next action. By continuing this process, the automaton learns to select an action with the best grade. The learning algorithm is used by automata to determine the selection of next action from the response of environment. Figure 2 shows the relationship between the environment and the learning automata [6].

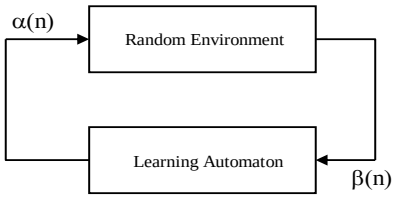

Figure 2: The relationship between learning automata and the environment

3.1 Environment

First, The environment can be shown by E ≡{α, β, c} in which α ={α_1,α_2,….,α_r} represents a finite action / output set, β={ β_1, β_2,…., β_m} represents an input / response set, and c={c_1, c_2,…,c_r} is the set of penalty probabilities, where each element c_i corresponds to one action α_i of the set α. The output (action) α_n of the automaton belongs to the set α, and it is applied to the environment at time t = n.

3.2 Learning Automata with Variable Structure

Variable structure learning automata is represented by < β, α, T, *p* >, where α = {α_1, α_2, …, α_r} is a set of actions. β = {0, 1} is the set of inputs from the environment; where 0 represents a reward and 1 represents a penalty, *p* (*n+1*) = T [α (*n*), β (n), *p* (*n*)] is learning algorithm and defines the method of updating the action probabilities on receiving an input from the random environment. *p* = {p_1 (*n*), p_2 (*n*)… p_r (*n*)} is the action probability vector, where $p_i(n)$ represents the probability of choosing action α_i at time *n*. In these kinds of automata, if the action of α_i is chosen in the n^{th} stage and receive the desirable response from the environment, the probability of $p_i(n)$ increases and the other probabilities decreases and in undesirable response, the probability of $p_i(n)$ decreases and the other probabilities increase. The following algorithm is one of the simplest learning schemes for updating action probabilities, and is defined as follows:

$$p_i(n+1) = p_i(n) + a[1 - p_i(n)]$$
$$\forall j \quad j \neq i \quad p_j(n+1) = (1-a)\,p_j(n) \qquad (1)$$

a) Desirable response

$$p_i(n+1) = (1-b)\,p_i(n)$$
$$\forall j \quad j \neq i \quad p_j(n+1) = \frac{b}{r-1} + (1-b)\,p_j(n) \qquad (2)$$

b) Undesirable response

As seen from the definition, the parameter *a* is associated with reward response, and the parameter *b* with penalty response. According to the values of *a* and *b* we can consider three scheme. If the learning parameters *a* and *b* are equals, the scheme called reward penalty (L_{R-P})When *b* is less than *a*, we call it linear reward epsilon penalty ($L_{R\varepsilon P}$) scheme. When *b* equals to zero, we call it as linear reward inaction (L_{R-I}) scheme. For more information about the theory and applications of learning automata, refer to [6] and [7].

4. Proposed Risk Management Algorithm

We proposed the inclusion in the ISO model of an optimization step between risk assessment and risk treatment steps aiming to automatically decide which risks have to be treated in order to satisfy objectives and needs of companies. Figure 3 indicates the position of the proposed optimization step inside the traditional risk management architecture. In this paper, the proposed optimization step has three main objectives:

1) Reducing total risk: selects a set of high impact and likelihood of risks to be treated;

2) Reducing total cost: Selects a set of risks that best fits to the available amount of financial resources for treating risks;

3) Reducing total time: Selects a set of risks that best fits to the available amount of time for treating risks.

Optimizing these three objectives leads to a set of risks to be treated that considers limited operational resources and minimizes company risk exposure time. The method used in the optimization step has to find risk treatments that promote reductions on the overall risk level and return a low level of residual risk. These objectives can be represented by the following three equations [5]:

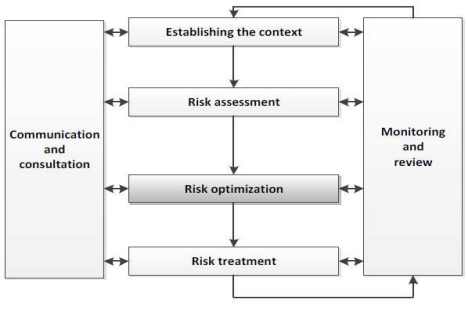

Figure 3: Proposed optimization step inside the risk management architecture

$$OveralRisk(R)=\sum_{r\in R}(RiskLevel_r-ResidualRisk_r) \quad (3)$$

$$OveralTime(R)=\sum_{r\in R}(TreatmentTime_r)-Time \quad (4)$$

$$OveralCost(R)=\sum_{r\in R}(TreatmentCost_r)-Cost \quad (5)$$

Where R is a set of risks selected in search space (e.g. output of the risk assessment step). The Time constant represents the amount of time reserved for treating risks. This constant can be estimated by summing the work-time of all workers. For instance, if the company has an operation team with 5 members and each worker works 6 hours per day, and then one week (5 workdays) of work contains 150 hours which can be used as the Time constant. The Cost constant represents the amount of money is available to be spent for treating risks over iterations. For accomplishing the proposed objectives the above mentioned equations are to be optimized using:

$$min[OveralRisk(R),OveralTime,OveralCost(R)]$$

This minimization step can be solved by several methods such as [11] and [4]. Other optimization algorithms based on Meta heuristics are plausible approaches [15] and [16] to tackle the above mentioned minimization problem due to their capabilities to deal with discrete and large search spaces. For this paper a solution based on Learning Automaton was suggested for composing the proposed optimization step. Assets and Risks play the role of stochastic environment in the learning automaton. In the proposed method for each Asset like $Asset_i$ where i=1,2,…,n the learning automaton of LA_i is considered. n shows the number of assets. r shows the number of actions in each automaton and α_{ij} In LA_i Automaton denote as a appropriated level of risk of j. Actions belong to each automaton such as LA_i considered as α_{i1}, $\alpha_{i2,\ldots,}$ α_{ir}.

Input Parameters:

Amount of assets, amount of risks per asset, amount of treatment per risk, vulnerabilities per asset, residual risk probability, new risks rate, new vulnerabilities rate, money budget, time window size، acceptable risk level and Reward/Penalty values;

Output: Optimize Total Risk, Total Cost, and Total Time and determine each risk level

1. For each $Asset_i$ consider Learning Automaton of LA_i

2. Set input parameters and initial value for each action considered 1/r

3. Determine initial values for OveralRisk(R),OveralTime(R) and OveralCost(R) by using these formulas:

$$OveralRisk(R)\leftarrow\sum_{i=1}^{NumberOfAsssets}\sum_{j=1}^{NumberOfRisks}(\alpha_{ij}\times Risk(i,j)\times Cost(i,j)\times Time(i,j))$$

$$OveralCost(R)\leftarrow\sum_{i=1}^{NumberOfAsssets}\sum_{j=1}^{NumberOfRisks}(\alpha_{ij}\times Cost(i,j))$$

$$OveralTime(R)\leftarrow\sum_{i=1}^{NumberOfAsssets}\sum_{j=1}^{NumberOfRisks}(\alpha_{ij}\times Time(i,j))$$

4.

$$OveralRisk_{Prev}\leftarrow OveralRisk(R)$$

5.

$$OveralCost_{Prev}\leftarrow OveralCost(R)$$

6.

$$OveralTime_{Prev}\leftarrow OveralTime(R)$$

7.`while`(`Overall risk is acceptable`) do

8.`for` i=1 to NumberOfAssets

8.1) Select Randomly One action such as j, where $1\le j\le NumberOfRisks$

8.2)

$$OveralRisk_{Cur}\leftarrow\sum_{i=1}^{NumberOfAsssets}\sum_{j=1}^{NumberOfRisks}(\alpha_{ij}\times Risk(i,j)\times Cost(i,j)\times Time(i,j))$$

8.3)

$$OveralCost_{Cur}\leftarrow\sum_{i=1}^{NumberOfAsssets}\sum_{j=1}^{NumberOfRisks}(\alpha_{ij}\times Cost(i,j))$$

8.4)

$$OveralTime_{Cur}\leftarrow\sum_{i=1}^{NumberOfAsssets}\sum_{j=1}^{NumberOfRisks}(\alpha_{ij}\times Time(i,j))$$

End of for.

if(OveralRisk_Prev>=OveralRisk_Cur and OveralCost_Prev>=OveralCost_Cur and TotalTime_Prev>=OveralTime_Cur **)**

Reward all of selected actions.

else

Penalize all of selected actions.

9.

$$OveralRisk(R) \leftarrow \sum_{i=1}^{NumberOfAssets} \sum_{j=1}^{NumberOfRisks} (\alpha_{ij} \times Risk(i,j) \times Cost(i,j) \times Time(i,j))$$

10.

$$OveralCost(R) \leftarrow \sum_{i=1}^{NumberOfAssets} \sum_{j=1}^{NumberOfRisks} (\alpha_{ij} \times Cost(i,j))$$

11.

$$OveralTime(R) \leftarrow \sum_{i=1}^{NumberOfAssets} \sum_{j=1}^{NumberOfRisks} (\alpha_{ij} \times Time(i,j))$$

12.

$$OveralRisk_{Cur} \leftarrow OveralRisk(R)$$

13.

$$OveralCost_{Cur} \leftarrow OveralCost(R)$$

14.

$$OveralTime_{Cur} \leftarrow OveralTime(R)$$

15.

$$OveralRisk_{Prev} \leftarrow OveralRisk_{Cur}$$

16.

$$OveralCost_{Prev} \leftarrow OveralCost_{Cur}$$

17.

$$OveralTime_{Prev} \leftarrow OveralTime_{Cur}$$

18. Treat selected risks and Merge residual risks.

19. End of While.

5. Experimental Results

For testing the proposed approach some main parameters are considered. They are depict in table 1: [5]

The virtual environment was generated by using the configuration showed in Table I. Our objective is to automatically find a subset of risks that maximize risk reduction and respect the company operational resource limitations. All possible combinations of found risks are represented by the power set of the risks returned by the simulator.

Table 1: Parameters

Parameter	Description	value
amount of assets	number of valuable objects related to the managed scope	100
amount of risks per asset	max number of risks for each asset	4
amount of treatment per risk	max number of possible treatments for each identified risk	2
vulnerabilities per asset	max number of found vulnerabilities per asset	2
residual risk probability	generation probability of new risk after the treatment process	0.02
new risks rate	rate of new risks to appear	0.002
new vulnerabilities rate	rate of new vulnerabilities to appear	0.002
Budget	Represents the amount of money is available to be spent for treating risks over iterations.	200
Time	represents the amount of time reserved for treating risks	400
acceptable risk level	Represents acceptable risk level	0.01
Reward/Penalty	Represents Reward and Penalty	0.03/0

Figure 4 shows evolution of the amount of risks per iteration over three different perspectives: (i) without risk treatment, (ii) simple treatment selection and (iii) optimized treatment selection. Risks are mitigated faster by using the optimized selection process. This simulated scenario promotes a reduction in the company risk exposure time. The “without selection” data series shows the simulation behavior without risk treatment (only the simulation). The “new risks rate” can be observed through the creation of new risks over iterations. These new risks are composed by identified risks and residual risks which are controlled by the “new risks rate” and “residual.

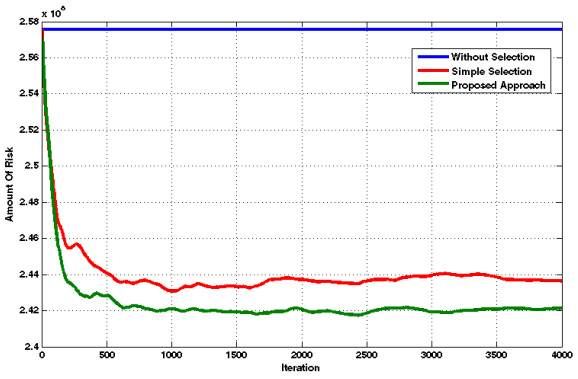

Figure 4:Amount of risks over iterations.

Figure 5 shows a comparison of total risk level over iterations between the simple and optimized selection approaches. The optimization algorithm has to select critical risks aiming the total risk level reduction of the analyzed scope.

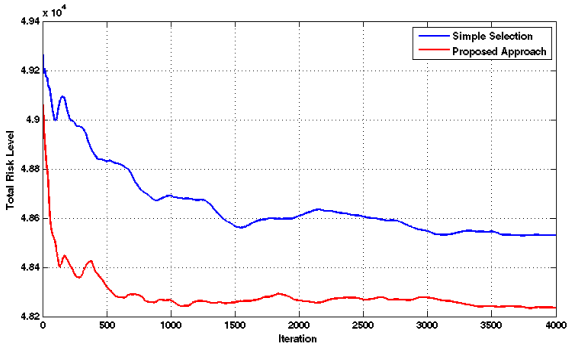

Figure 5: Total risks Level over iterations.

Risk treatment selection using the proposed approach reaches an acceptable risk level 20% faster than the simple selection approach. This improvement represents a huge gain of resources for the analyzed company. Saved resources along iterations can be human resources, time or financial resources. Beside all that, the company total exposure time is reduced and the probability of occurrence of critical incidents on the analyzed scope is reduced. Figure 6 shows a comparison between the simple selection approach and the optimized approach on instantaneous time spent over iterations. We can observe that the optimized selection better uses the time available per iteration. The proposed solutions are closer to the optimal solution hence; the operational team is more successfully used and has less idle time.

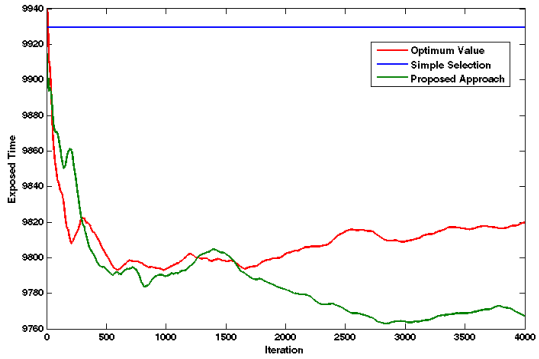

Figure 6: Exposed time over iterations.

This document is set in 10-point Times New Roman. If absolutely necessary, we suggest the use of condensed line spacing rather than smaller point sizes. Some technical formatting software print mathematical formulas in italic type, with subscripts and superscripts in a slightly smaller font size. This is acceptable.

Figure 7 shows a comparison between the simple selection approach and the optimized approach about the financial resource application over iterations. The optimized selection better uses financial resources available over iteration. The optimized selection approach proposes solutions closer to the optimal solution (e.g. iteration money budget size) than simple selection solutions. This objective is less flexible than the time window filling objective once the budget for treating risks is, most of time, controlled by non-technical departments inside the company (e.g. financial and accountability departments).

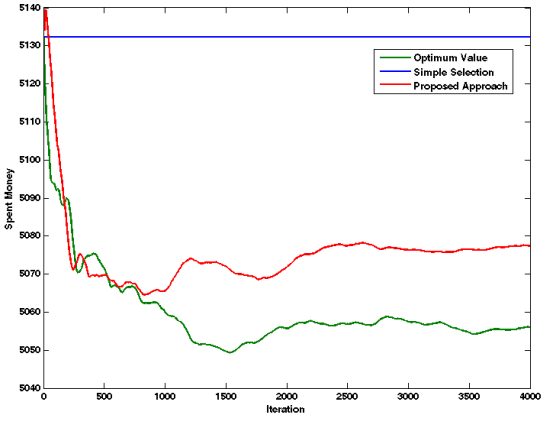

Figure 7: Spent money over iterations.

6. Conclusion

This paper proposes an optimization step for the traditional risk management process which is responsible for selecting risks to be treated aiming at optimizing the organization objectives. For validating the proposed approach, a simulated environment was created which has assets, generates a risk and receives risks to be treated. An information security scenario was created for testing the proposed optimization module. Finally, due to the conflicting (objectives) and combinatorial nature of the treated problem a mulch-objective Learning Automaton was used. Experiments show that the optimized approach guides the simulated company to an acceptable risk level, in average 20% faster than a traditional approach. This means that operational resources are better used and the company risk exposure time is reduced, protecting the treated scope against threats and eventual losses. Technical team was more efficiently used, once they had less idle time. Experiments also show that the proposed approach can be scaled for analyzing large scopes that contains more assets. Although, the performed experiments were used for managing risks related to information assets aiming increase their security level, the proposed approach can be applied for managing risks related to any type of assets.

References

[1] I. O. for Standardization, "Risk management - Principles and guidelines", ISO 31000, Oct. 2009.

[2] ——, "Risk Management - Risk assessment techniques", ISO 31010, 2009.

[3] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan, "A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II ", in Parallel Problem Solving from Nature PPSN VI. Springer, 2000, pp. 849-858.

[4] J. Durillo, A. Nebro, and F. Luna, "Convergence speed in multiobjective metaheuristics: Efficiency criteria and empirical study ", International Journal, May.2010, pp. 1344-1375.

[5] Marcos Álvares Barbosa Junior, Fernando Buarque de Lima Neto and Tshilidzi Marwala. "Optimizing risk management using NSGA-II", *in* 2012 IEEE Congress on Evolutionary Computation (CEC'2012), IEEE Press, Brisbane, Australia, June 10-15, 2012, pp. 1325-1332.

[6] M. R. Meybodi, and S. Lakshmivarahan, "On A class of Learning Algorithms Which have Symmetric Behavior under Success and Failure", Lecture Notes in Statistics, Berlin: Springer Verlag, 1984, pp. 145-155.

[7] K. S. Narendra, and M. A. L., Thathachar, Learning Automata: An introduction, Prentice Hall, 1989.

[8] R. T. F. A. King, H. C. S. Rughooputh, and K. Deb, "Solving the mul-tiobjective environmental/economic dispatch problem with prohibited operating zones using NSGA-II", in Proceedings of 2011 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing. IEEE, Aug 2011, pp. 298-303.

[9] X. Jiang, C. Yongqiang, W. Xiaoqing, Z. Minhui, and X. Liu, "Optimum design of antenna pattern for spaceborne SAR performance using improved NSGA-II", in 2007 IEEE International Geoscience and Remote Sensing Symposium IEEE, 2007,pp.615-618.

[10] J. Jia, J. Chen, G. Chang, J. Li, and Y. Jia, "Coverage Optimization based on Improved NSGA-II in Wireless Sensor Network" ,in 2007 IEEE International Conference on Integration Technology. IEEE, Mar. 2007, pp. 614-618.

[11] K. K. Mishra, A. Kumar, and A. Misra, "A variant of NSGA-II for solving priority based optimization problems ", in 2009 IEEE International Conference on Intelligent Computing and Intelligent Systems, vol. 1. IEEE, Nov. 2009, pp. 612-615.

[12] P. Murugan, S. Kannan, and S. Baskar, "Application of NSGA-II Algorithm to Single-Objective Transmission Constrained Generation Expansion Planning", IEEE Transactions on Power Systems, Vol. 24, No. 4, Nov 2009, pp. 1790-1797.

[13] S. Mishra, G. Panda, S. Meher, R. Majhi, and M. Singh, "Portfolio management assessment by four multiobjective optimization algorithm", in 2011 IEEE Recent Advances in Intelligent Computational Systems. IEEE, Sep. 2011, pp. 326-331.

[14] I. Das and J. E. Dennis, "Normal-Boundary Intersection: A New Method for Generating the Pareto Surface in Nonlinear Multicriteria Optimization Problems", SIAM Journal on Optimization, Vol. 8, No. 3, Jul. 1998, pp. 631.

[15]A. Colorni, M. Dorigo and V. Maniezzo, "Distributed Optimization by Ant Colonies", Proceedings of the First European Conference on Artificial Life, Paris, France, Elsevier Publishing, 1992,pp.134-142.

[16]C. Coello, G. Lamont, and D. Van Veldhuizen, "Evolutionary algorithms for solving multi-objective problems", Springer-Verlag, New York, vol. 5, 2007.

Babak Anari received the B.Sc. degrees in Software engineering from Azad University of Shabestar, Iran in 2002, and M.Sc. degrees from Azad University of Arak, Iran in 2007, respectively. He is currently a PhD Student in Islamic Azad University, Science and Research Branch,Tehran,Iran. He has many research papers in learning Automata and web Mining Fields. His research interests include learning systems, parallel algorithms, soft computing, Distributed Systems and software development.

Mohammad Reza Ahmadi received the B.Sc. and M.Sc. degrees in Electrical Engineering and Communication Systems from K.N.T. University of Technology in 1986 and 1990 respectively. He received his Doctor degree in Communication Networks from Tokyo Institute of Technology, Tokyo, in 1997. Currently he is the project manager and researcher in IT department of Research Institute of ICT (CSRI). His research interests are network security focus on intrusion detection/prediction systems, Immune systems focus on

network applications and data center design and implementation.

Mostafa Ghobaei Arani received the B.Sc. degrees in Software engineering from Azad University of Kashan, Iran in 2009, and M.Sc. degrees from Azad University of Tehran, Iran in 2011, respectively. He is currently a PhD Student in Islamic Azad University, Science and Research Branch,Tehran,Iran He has many research papers in Grid Computing and Cloud Computing Fields. His research interests include Grid Computing, Cloud Computing, Pervasive Computing, Distributed Systems, and software development.

Zohreh Anari received the B.Sc. and M.Sc. degrees in software engineering from Islamic Azad University of Shabestar, Iran in 2003 and 2009, respectively. She has many research papers in fuzzy and web mining fields. Her current research interests include learning automata, web mining and soft computing. Her research interests include learning systems, parallel algorithms, soft computing, and software development.