Modeling Cooperation between Nodes in Wireless Networks by APD Game

Abbas Arghavani[1], Abdorasoul Ghasemi[2]

1) Computer and Information Technology Department, Qazvin Islamic Azad University, Qazvin, Iran
2) Faculty of Electrical and Computer Engineering Department, K. N. Toosi University of Technology, Tehran, Iran

a.arghavani@qiau.ac.ir; arghasemi@kntu.ac.ir

Abstract

   Cooperation is the foundation of many protocols in wireless networks. Without cooperation, the performance of a network significantly decreases. Hence, all nodes in traditional networks are required to cooperate with each other. In this paper, instead of traditional networks, a network of rational and autonomous nodes is considered. Which means that each node itself can decide whether to cooperate with its neighbor or not and performs something that benefits it. We have used Alternative Prisoner's Dilemma game which is one of the classic games in the field of game theory, to model node's behavior in a nontraditional network. Then, by providing an approach based on Learning Automata, we've tried to encourage the nodes to cooperate with each other.

Keywords: *Wireless Network, Cooperation, Game Theory, Prisoner's Dilemma, APD Game*

1. Introduction

Environmental destructive phenomena such as fading, shadowing and blocking, have harmful effects on the performance of wireless networks. Up to the present time, a variety of solutions called diversity, have been proposed to overcome these destructive phenomena. One of the most popular methods of diversity is called cooperative diversity [1]. In this method, if a direct connection between two nodes is not possible for any reason (such as fading, shadowing or blocking), these nodes can communicate with each other through their neighbors. For further explanation, consider Figure 1. Suppose that node S intends to send a data packet to node D_s. However, because of the obstacle between two nodes, it is not possible to send directly. Cooperative diversity suggests that node R acts as a relay node, which means that it receives the data packet sent from S and forwards it toward D_s.
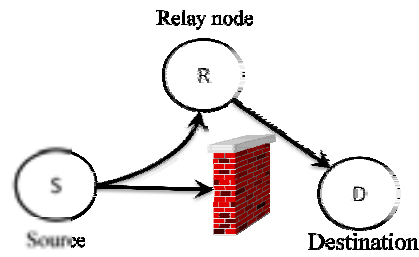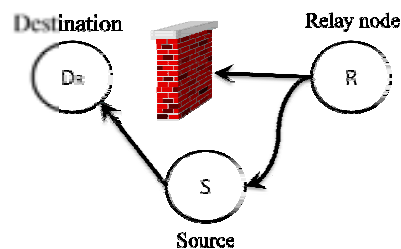
Figure1. Node S needs node R's cooperation

As we know, in a traditional network, node R had to cooperate with node S. Modeling and monitoring the behavior of autonomous nodes in a nontraditional network are our main goal in this paper. A nontraditional network comprises autonomous nodes which behave rationally. An autonomous node is a node that independently decides about its own action in every status. It should also be known that a rational node is a selfish node chooses the action which is more beneficial for it from the possible actions (cooperate or non-cooperate). In other words, a rational node prefers its own profit to network's gain.

The main challenge in such network is the cost of cooperation. It is obvious that helping node S incurs costs to cooperative node R. This cost is a function of power consumed in forwarding the packet of node S. Although it will be very little but node R, which we suppose it as an autonomous and rational node, will be discouraged from cooperating.

According to this, apparently in a nontraditional network, nodes do not cooperate with each other because it is inconsistent with the assumption of being a selfish node.

On the other hand, in consequence of high volume of interactions between nodes in a network, there can be a situation in which node R requires node S's assistance. Therefore, node S will retaliate node R's non cooperation. Consider Figure 2. In this scenario, for similar reasons, forwarding data packets from node R to D_R is not possible directly. Now it has been expected that node S compensates node R's behavior in the previous scenario. But, decision making about cooperating or not, is not that simple for node S because the scenario explained in Figure 1 may happen again and it is forced to use node R's assistance.

Figure 2. Node R needs node S's cooperation

In this paper, the basic assumption is that the considered network includes a set of rational and autonomous nodes which always performs an action that benefits them. In a wireless network in which cooperation is the foundation of many protocols, this

behavior will decrease network's performance. Accordingly, the purpose of the proposed approach is summarized as follow:

- Modeling the behavior of nodes in their interactions with each other (cooperation and noncooperation) by using game theory and APD game.
- Using L_{RP} learning automata to encourage the nodes to cooperate with each other.

Game theory is a bag of analytical tools designed to help us understand the interaction between several decision makers. The basic assumptions that underlie the theory are that decision makers pursue well defend objectives because they are rational and take into account their knowledge or expectations of other decision maker's behavior which indicates that they strategically reason [2].

Section 2 is devoted to describe the APD game and how it is used to model the behavior of nodes. In section 3, the evolutionary games will be reviewed. These games that have been inspired from Darwin's theory of evolution are helpful to create a solution for encouraging the nodes to cooperate. Evaluation of several proposed solutions in APD, in the field of social and economics science, is surveyed in section 4. Section 5 will present a learning automata based approach whereby we can encourage the nodes to cooperate with each other. Section 6 is dedicated to the simulation results. Section 7 concludes the paper while identifying some potential future works.

2. APD Game

The Prisoner's Dilemma game considers the behavior of two thieves that have partnered in a robbery and now have been arrested by police. Since there isn't enough evidence to prove their crime, they are interrogated. Each thief should choose an action between confessing and keeping silence. In this paper, Confession and silence are respectively considered as cooperation (C) and noncooperation (D) and the term player is used instead of the term prisoner. The outcome of Prisoner's Dilemma game is presented in the following payoff table (Table 1).

Table1. The payoff table of Prisoner's Dilemma game

	D	C
D	P, P	T, S
C	S, T	R, R

According to table 1, if two players cooperate with each other, both of them will benefit the value of R. If both betray each other and do not cooperate, they will benefit the value of P, and finally if one of them cooperates and the other one does not cooperate, they will respectively profit the values of S and T. The relationship between values of R, P, T and S are as follow:

- T>R>P>S
- $2 \times R > (T + P)$

If the above game is once played between two players, it is obvious that they will prefer noncooperation. In game theory, such finality in making decision is called Nash Equilibrium [3]. The situations of two players in this game are very similar to those described in Figure 1 and 2, in which two rational nodes should decide to cooperate or not. Note that in wireless networks, two neighboring nodes may stand in such situation

several times. Consequently, the evolved type of Prisoner's Dilemma game known as APD should be reviewed.

APD game is the changed kind of Iterative Prisoner's Dilemma (IPD), in which, players execute an iterative game and simultaneously decide about their actions ("C" or "D"). While in APD, this decision making process is performed intermittently. The profit of one player will be depended on its own choice and the decision of its opponent in previous game. So, the total profit of one player in one round of APD game consisted of n Prisoner's Dilemma game, is calculated from the sum of profits in each game. But, in wireless networks, what does this profit mean?

It has been known that a node, like A, consumes energy for cooperation with another node, like B. So, cooperation is equal to loss. The value of loss is shown with β. On the other hand, node B will be benefited because its packet is directed toward the target and it no longer needs to retransmit it. The value of payoff is shown with α. So nodes cooperation game is shown with Table 2 as follow.

Table 2. Normalized profit matrix

	D	C
D	0, 0	α, -β
C	-β, α	α – β, α – β

For simplicity, it is assumed that APD game is held in a network consists of a pair of nodes, A and B, but following explanations can be applied to a network with n nodes.
Node A's actions is shown with "C" and "D" and node B's actions is shown with letter cases, "c" and "d". So a portion of action string can be like "DcCd…"
In order to calculate each node's payoff, the action string of two nodes is examined character by character from beginning to end. For each action in the action string, previous game is also considered and then according to Table 2, we calculate each player's payoff. For example, assume that nodes A and B are behaved like "DcCd" which means that in the first game, node A dose not cooperate with node B ("D"). Since this is the first game and there is not a previous game, so calculation of the payoff is not possible. Then, node B cooperates with A ("c") and the packet is forwarded from A toward the target. Now, the privilege of this choice can be calculated. This action and the previous one make a couple of actions ("Dc"). Thus, based on Table 2, node A and B respectively gain the value of α and –β. The third action done by node A is cooperation. For calculating the payoff, two last actions ("cC") are considered. Thus, each node gains the value of α – β. In next step, node B does not cooperate ("d") and gains the value of α and Node A will gain –β.
In a computer network, the size of an action string will be much larger because two nodes frequently need to make decisions whether to cooperate together or not. The payoff of a node during an action string can be obtained from the sum of payoffs from every single game. Each node that gains more will be considered as the winner which means this node's strategy is better for playing in a nontraditional network. A strategy defines that how a node acts in different situations and what decisions should be made. Some proposed strategies in the field of APD game are represented in section 4.

3. Evolutionary Games

Evolutionary games have been applied most widely in the area of evolutionary biology, the domain in which the idea was first articulated by J. M. Smith [4]. Evolutionary game has been based on the idea that a strategy that fits more, will tend to produce more offspring.

Suppose that there are M strategies in an environment. At generation *n*, each strategy is represented by a certain number of players as $W^n(S_i)$. S_i is one of the M strategies and $V(S_i|S_j)$ is the score of S_i strategy when it plays with S_j strategy. This score is calculated from Table 1. Relation (1) is used for computing each strategy's score. $g^n(S_i)$ is the score of S_i strategy at the end of generation *n*.

$$g^n(S_i) = \sum_{j=1}^{M}\left(W^n(S_j)V(S_i|S_j)\right) - V(S_i|S_i) \tag{1}$$

The size of S_i strategy in generation *n + 1* ($W^{n+1}(S_i)$) is determined by its score in generation *n*.

$$W^{n+1}(S_i) = \left\lfloor \frac{W^n(S_i)g^n(S_i)\sum_{j=1}^{M}W^n(S_j)}{\sum_{j=1}^{M}(W^n(S_j)g^n(S_j))} \right\rfloor \tag{2}$$

4. Known Strategies

A strategy is an algorithm that determines what should be done by each player in any game. The objective of each strategy is to get more points for its player. So far, many strategies have been proposed to play in APD and TFT strategy is the most popular one. TFT (Tit For Tat) is probably the most-studied strategy in game theory [5]. A player using this strategy will initially cooperate, and then respond in kind to an opponent's previous action. If the opponent was cooperative previously, then the player will cooperate and otherwise, it won't. TFT submitted by AnatolRapaport in Axelrod's Tournament in 1984 [6].
To learn more about this strategy, two nodes A and B are considered. This action string could be created by A and B: "C,c,C,d,D,…". Assume that node A will play the TFT strategy so it cooperates in first game. Node B answers with cooperation based on its own strategy. From now on, in any game, node A performs the same action done by node B in previous game, so it cooperates in the third step and this process continues. In [6], this extremely simple strategy scored higher than other strategies and took the first place.
ALLC strategy (Always Cooperate) is a strategy that cooperates in all games and ALLD strategy (Always Defect) is a strategy that defects in all games. Random strategy selects one of the actions (cooperation or defection) with equal possibility in every game.

5. Optimality in APD

It is much less obvious as to what an optimal strategy would be or even how an optimal strategy should be defined [7].

In the APD, determining circumstances which make an optimal strategy is very difficult. As it has been noted the performance of a strategy is highly dependent on other strategies it interacts with. This has been led to several conflicting definitions of

the term "optimality", with resulting differences in which strategies (if any) are considered optimal. According to this, optimality has been defined relative to a given set of opponents: the optimal strategy is the one achieves the highest score (with respect to some measures) against that set of opponents. One typical measure of performance is the average score in a round-robin tournament interaction. In reality, there is no fixed strategy performs best against every given set of opponents in this interaction because achieving optimality needs forecasting opponents action before their act and this is impossible [7]. So the considered problem in this paper is to find a suboptimal strategy which will ensure victory in packet relaying tournament.

A good strategy is the strategy which stays alive in the population for the longest possible time and in the biggest possible proportion [8]. In evolutionary games, such strategy is called evolutionary stable strategy (ESS). A strategy is an ESS if, when the whole of population is using this strategy, any small group of invaders using a different strategy will eventually die off over multiple generations [9].

One strategy can be suboptimal provided that it has four characteristics which have been explained in [6] and [10]. These characteristics are:

- It should to be good (it starts by cooperating)
- It should to be retaliating (it returns the opponent's defection)
- It should to be generous (it forgets the past if the defecting opponent cooperates again)
- It should to be not memory less (it utilizes the interaction history)

6. Proposed Strategy

As mentioned before, it is impossible to reach the optimal strategy, but expectation of finding a way which can estimate the behavior of the opposite strategy is not illogical. Learning is a process that living beings need it for making changes in their behavior and being compatible with the environment. Stochastic learning automata [11] is a decision making algorithm that acts in a stochastic environment and updates its strategy for the next action based on the response that it gets from the interaction with the environment.

Learning automata doesn't know the environment at the beginning of the game, there for it tries to know the environment with trial and error. So, at first it performs actions randomly. The environment of the game (here is the opposite strategy) reacts in front of each action and learning automata reduces or increases the possibility of performing this action based on the amount of desirability or undesirability of it before the next game begins.

The environment of the game is defined by (α, c, β). α is the sum of actions that can be done by the automata. It's evident that in the mentioned environment, α equals $\{C, D\}$ which "C" means cooperation and "D" means noncooperation. Set c is the possibility of penalty and each member in this set represents the possibility of receiving penalty provided that the action is done from the set of actions. Determining set β is in proportion with the amount of penalty and the reward which has been inserted in Table 1.
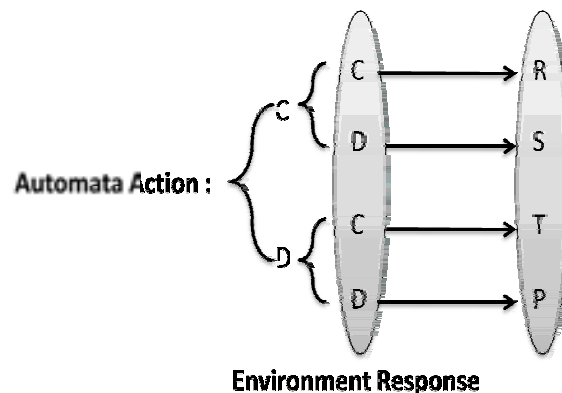
Figure 3. Considering the environment response to the behavior of automata

Based on Table 1, the environment's reactions to the behavior of the automata is $\beta = \{S, P, R, T\}$ and the amounts in the interval of [0, 1] are allocated to members of this set. For this reason, the environment's reactions are divided by X in which $X = S + P + R + T$. So, the set of the environment's reactions is defined as $\beta = \{\frac{S}{X}, \frac{P}{X}, \frac{R}{X}, \frac{T}{X}\}$. In continuation, an indicator is required for determining the amount of desirability of the environment's reactions. It has been known that if two nodes make effort to delete the packets of each other, they have neither profit nor lose (obtaining score P). A node will profit when his opposite node directs his packet (obtaining score R or T). So a desirable action is the one that the amount of its score which has been obtained from environment's reaction is more than P and an undesirable action is the one that the amount of its score is less than P. The amount of desirability and undesirability is dependent on the difference of the score which has been obtained of or from P.
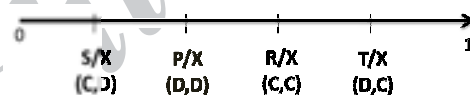

Figure 4. Evaluating the desirability of the environments response

Considering that, the number of members in set β have been limited and kept in the interval of [0, 1], the model of the game's environment is Q-Model. We have used LRP automata for updating the possibility of performing the actions of the set of automata's actions because an automata will be victorious that rewards or fines its action with every reaction receives from the environment, and only in such situation, the possibility of confronting to malevolent strategies like ALLD is provided. If P_C and P_D are respectively considered as the possibility of performing function "C" and "D", then the following pseudo code (Figure 5) shows the trend of updating each above possibilities. "i" is the action which is performed by an automata in the nth game ($i \in \alpha$) and $-i$ represents the other actions of the member of set α.

7

```
if (i action had a good response in previous game)
        P_i-next = P_i-prev + Reward × (1 – P_i-prev)
        P_-i-next = (1-Reward) × P_-i-prev
else
        P_i-next = (1 - Penalty) × P_i-prev
        P_-i-next = Penalty + (1 - Penalty) × P_-i-prev
end
```

Figure5. Learning automata Pseudo code (LRP)

The important problem in this approach is determining the values of reward and penalty. As mentioned before, the desirability of received reactions determines the value of reward or penalty which has been allocated to each performed action. Table 2 shows the amounts of reward or penalty of each performed action.

Table 3. Determining the value of penalty and reward

Action	Bad Response	Good Response
C	(S - P)/x	(R - P)/x
D	(P - P)/x	(T - P)/x

7. Simulation results

Around the world, there are various strategies for living known as ideology. Each strategy claims to provide the best way for living that brings happiness to its followers. Reasonable people are looking for happiness and trying to find the best strategy for living. People, intentionally or unintentionally, follow Darwin's theory of evolution. Each person, based on his cognitive, selects one of the existing strategies for a while. If this strategy satisfies this person, it will be retained otherwise it will be left and another strategy is chosen by this person. Accordingly, if all people in the world are supposed to be rational, after a while the best way of life is hopefully found by each person.

7.1 Finding the best strategy in noise-free environment

In this paper, the Darwin's theory of evolution is used to find the best strategy for playing in the APD game. For this reason, the concept of noise in the APD game is initially represented. Noise in the APD game is different from noise in the network. When an action (cooperation or noncooperation) is chosen by a node, the opposite node will be aware of this behavior. This information is obtained through a global or distributed mechanism. So, there is a notification mechanism that informs each node of its neighbor's behavior. If this notification mechanism is an error free mechanism; that is the behavior of each node is reported correctly, then the environment will be a noise-free environment. The simulation results for noise-free environment are analyzed as follow.

A fully connected network of 60 rational nodes is considered since the network topology between nodes, packet size, the amount of energy in each node, and so on, do not influence on a node's behavior for playing in APD game. But the important thing is

8

the amount of a node's interactions which indicates the number of times this node will enter the APD game. When two nodes play the APD game together, their strategies compete to earn more profits. Thus, whatever the Interactions of a node are greater, the quality of strategy that is used, will be more obvious. In order to select the best strategy from five strategies, TFT, ALLC, ALLD, Random and LA, a network with 50 nodes is considered. At first, none of the nodes know which strategy is better, so they choose one of the above strategies with equal probability. For fairness, it is assumed that in the first generation of games, each strategy is chosen by ten nodes. That is, in the first generation of APD game, each strategy has 10 follower nodes. Our intention of generation is one round of APD game that is held between nodes. Within a generation, the strategy of each node is fixed, but at the end of it, a node can change its strategy and continue the game with a new strategy. Since the network is fully connected, each node makes an APD game with every 59 other nodes. This means that every strategy will compete with all of other strategies (like a football league). The length of APD game in each generation is considered as 4000 game. This number does not match a specified criteria and it should be enough so that the impact of a game's score is insignificant in final score.

Nodes play a round together based on their own strategies and at the end of the game each node calculates its score. Each node's score is the sum of the points earned in competition with every other node. After the nodes calculated their points, it's time to calculate the score of each strategy. The score of each strategy is obtained from the sum of node's scores using this strategy (Equation1).

Now the Darwin's theory of evolution is used. This simple law states that the number of followers of a strategy for next generation is consistent with the scores that has been obtained in the current generation (Equation 2). For example, if two strategies, x and y respectively obtain 100 and 50 scores then the number of nodes in next generation that use the x strategy will be doubled of the number of nodes that use the y strategy.

This game is held for successive generations and at the end of each generation, nodes can change their strategies. The game ends when no nodes are willing to change their strategies. Figure 6 shows the simulation results. The horizontal axis shows the passing generations, and vertical axis shows the number of nodes in each strategy. It is obvious that the behavior of two strategies, TFT and LA are quite similar. While the generation of two strategies, ALLD and Random are entirely extinct. This means that nodes gradually realize that LA and TFT strategies are more profitable than other strategies, but by using strategies Random and ALLD, substantial profits are earned. Hence, gradually these strategies have been excluded by logical nodes.
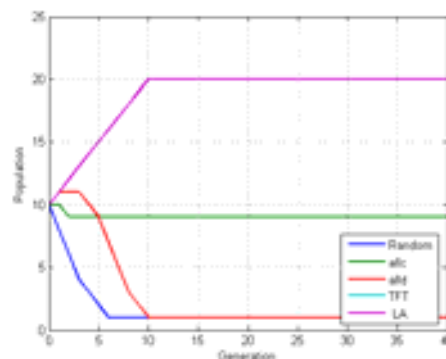
Figure 6: Simulation Results in a Noise Free Environment

9

This is a reasonable result because strategies with desiring to cooperate have been remained in the environment. Three strategies, ALLC, TFT and LA prefer to cooperate. As a result, in a nontraditional network, with three strategies, ALLC, TFT and LA (and without regard to other strategies), nodes prefer cooperation to non-cooperation.

These above experiments have been tested for more strategies and in all cases, cooperative strategies encourages nodes to follow them.

7.2 Finding the best strategy in noisy environments

The only difference between a noise free environment and a noisy environment is the existing of some error in the notification mechanism. For example, node A may cooperate with node B, but the reverse of this (noncooperation) is notified to node B. So, node B thinks that node A is not cooperative. Noise can cause Domino Effects. This phenomenon is a long chain of unwanted events with small sources and is the main reason of declining the population of good strategies, so the tendency of nodes for cooperation in noisy environments will decrease. Consider the following example.

One of the strategies is strongly influenced by Domino Effect is TFT strategy. Suppose two nodes play together with TFT strategy. Table 4 shows the game between two nodes.

Table4. The game between two nodes playing with TFT strategy in a noisy environment and occurring Domino Effect

Stage number	1	2	3	4	5	6	7	…
Action String	C	c	C	d	D	d	D	…

As seen in Table 4, two nodes A and B, initially cooperate with each other, but in the third game, the notification mechanism has been failed and a wrong information of node A's behavior is given to node B; cooperation is reported as noncooperation. Considering the nature of the strategies used in the two nodes (TFT strategy), node B immediately reacts and does not cooperate with node A. So, when node A observes this action, it retaliates and a chain of retaliation will occur.

Figures 7, 8 and 9 show the simulation results for noises with the amount of 1%, 5% and 10%. It can be seen that there is no resistance strategy that works well in noisy environments and encourages the nodes to cooperate with each other. Simulation is also performed for more noises. With increasing noise, cooperative strategies are affected more than other strategies, so that in high amount of noise all rational nodes are inclined to ALLD strategy. This means that, when nodes do not have correct information about the behavior of the other nodes, they prefer noncooperation.
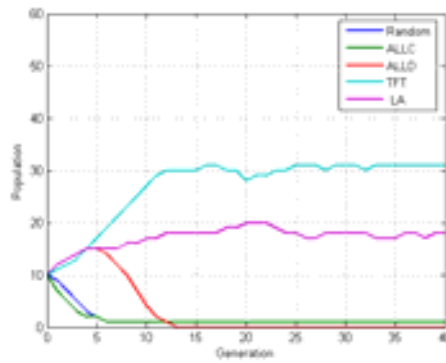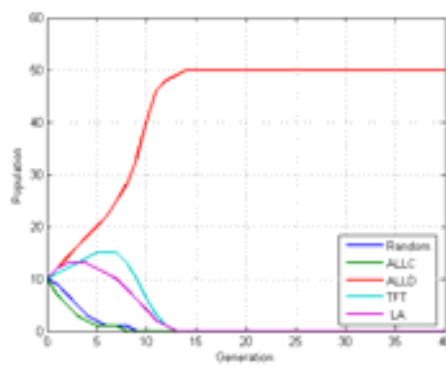
Figure 7: Simulation Results (Noise Ratio = 1%)

Figure 8: Simulation Results (Noise Ratio = 5%)
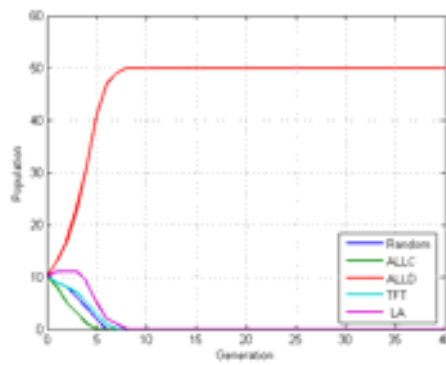

Figure 9: Simulation Results (Noise Ratio = 10%)

11

8. Conclusion

In this paper, by using APD game which, is one of the classic games in game theory, we have examined the behavior of rational nodes to find out whether they are cooperating with each other or not. Then, the TFT strategy that is one of the most popular methods to play in APD has been described and besides providing a learning automata based approach, it is shown that in a noise free environment, nodes can be encouraged to cooperate together.

Unfortunately, in noisy environments, due to uncertainties in the behavior of a node, which is caused by the abnormalities of the notification mechanism, no approaches were found to encourage the nodes to cooperate with each other. In other words, in a noisy environment, rational nodes prefer noncooperation to cooperation. What can be considered as next step is examining the effect of increasing and decreasing the amount of gain or loss included in the payoff table on the convergence process of strategies. Our intention of convergence is the time required by the nodes of a network for finding a superior strategy.

9. References

[1] J. N. Laneman, "Cooperative Diversity in Wireless Networks: Algorithms and Architectures," P.HD thesis, Faculty of Electronic, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, 2002.

[2] G. Abdoli, *Games theory and applications,* Jahad daneshgahi publications, Iran, 1390.

[3] J. F. Nash, "Equilibrium Points in n-Person Games," *in Proceedings of the National Academy of Sciences*, Washington, DC, PP. 47-49, 1950.

[4] Smith, J. M., 1982. Evolution and the Theory of games, Cambridge University Press.

[5] G. J. Mailath et al., *Repeated games and reputations: long-run relationships,* Oxford university press, USA, 2006.

[6] Axelrod, R., 1984. The Evolution of Cooperation, Basic Books.

[7] B. Neill, "Optimality Under Noise: Higher Memory Strategies for the Alternating Prisoner's Dilemma," *Journal of Theoretical Biology*, vol. 211, no. 2, pp. 159-180, 2001.

[8] Wu, J., et al., 1995. How to Cope with Noise in the Iterated Prisoner's Dilemma, Journal of Conflict Resolution, vol. 39, no. 1, pp: 183-189.

[9] J. N. Webb, "*Game Theory: Decisions Interaction and Evolution*," First ed., Springer, 2006.

[10] L. Yan et al., "Cooperative Packet Relaying Model for Wireless Ad Hoc Networks," *IEEE/ACM International Workshop on Foundations of Wireless Ad hoc and Sensor Networking and Computing (FOWANC' 08)*, Hong Kong, China, pp. 93-99, 2008.

[11] S. N. Sivanandam et al., "*Introduction to Genetic Algorithms*," Springer, 2007.