On the Nash Equilibria of Graphical Games for Channel Access in Multihop Wireless Networks

Vaggelis G. Douros, Stavros Toumpis, George C. Polyzos
Mobile Multimedia Laboratory, Department of Informatics
Athens University of Economics and Business
Patision 76, 10434, Athens, Greece
E-mail:{douros, toumpis, polyzos}@aueb.gr

***Abstract*—We study a multihop wireless network with a tree topology where selfish nodes compete for channel access. Firstly, we discuss how this is relevant to future multihop networks focusing on the concept of Device-to-Device communication. We then model this framework as a graphical game, which is a special case of a non-cooperative game in which the payoff of a node is influenced only by a subset of its neighbors. We discuss two payoff models that may be used depending on the application: The first focuses only on transmitters and assigns by default a zero payoff to the receivers and the second models a non-zero payoff to both transmitters and receivers. We then present a distributed scheme that finds an efficient Nash Equilibrium (NE) under both payoff models. We evaluate its performance through extensive simulations showing that the algorithm converges fast to a NE, in a number of rounds that is proportional to the logarithm of the number of nodes. Finally, we find that the number of successful transmissions is almost equal at any NE. This indicates that, under this metric, any NE is practically equally preferable.**

I. Introduction

Mobile data traffic, especially mobile video traffic, has dramatically increased in recent years with the emergence of smart phones and tablets. A major issue in future cellular systems is to make high bit rates available to a larger portion of the cell, especially to users in exposed positions in between several base stations. Also, regardless of their position, the nodes that transmit data should share efficiently the limited radio spectrum. When a node transmits, other nodes are affected. Therefore, it is critical to define rules on how the nodes should share the radio channels.

There have been various multiple access methods proposed that allow multiple nodes to share a common channel when they transmit. Such multiple access schemes can be classified as either contention-free channel access (e.g., Frequency Division Multiple Access) or contention-based random access methods (e.g., Aloha). In a multiple access scheme, nodes can either compete or cooperate so that either an individual or a group objective can be achieved. For this reason, the framework of game theory has recently become a very useful mathematical tool for modeling and analyzing multiple access schemes in wireless networks [1].

In this paper, we study contention-based channel access in the context of multihop wireless networks. Our work is general enough to be applied to any type of multihop wireless network. However, our emphasis is on future multihop networks. For example, our work is relevant to the Device-to-Device (D2D) communication [2], which is expected to be a key technology in 5G systems for providing seamless, high quality wireless access. D2D are expected to extend internet access services and support a number of emerging applications (disaster recovery scenarios, applications of vehicular ad hoc networks and wireless sensor networks). The D2D concept is built around the idea of allowing wireless devices to communicate with one another via direct D2D links over both the licensed and unlicensed spectrum. Therefore, traffic can be offloaded from the core networks, better service is provided to users, and both cellular coverage and energy efficiency are improved. Efficient schemes for channel access are a prerequisite for a successful deployment of D2D communication systems.

In this setting, solving the problem of multiple access through a centralized scheme imposes a significant communication and computation overhead that increases significantly with the network size. In contrast, efficient distributed algorithms can be designed based on non-cooperative game theory that are neither computationally expensive, nor increase network overhead. In this case, each node chooses its strategy selfishly, without any a priori coordination with the other nodes, aiming at maximizing its individual payoff. We model this multihop wireless network as a graph, focusing on tree topologies. We assume that nodes want to transmit their packets only to other nodes that are 1 hop away, *i.e.*, their immediate neighbors. Many interesting scenarios fall into this category. For example, all the devices that a person carries with him and his cell phone, which can be considered one layer up when it can be connected to an access point or another type of relay on the same channel. Moreover, the devices may be far from each other or tune their power to connect only to the cell phone, making direct transmission and loops impossible. Finally, the access point might be connected to other cell phones in a similar way, leading to a 2-level tree topology. If the access point is not fixed, but a mobile device, it could be part of an ad hoc network, possibly with a tree topology, leading to deeper trees.

Our contributions are three-fold: (i) We model this setup as a special case of a non-cooperative game called graphical game. Contrary to a typical non-cooperative game, in a graphical game the payoff of the nodes depends only on the strategies of (some of) their neighbors, in this case those that are up to 2 hops away. (ii) We analyze the game under two payoff models (with and without modeling a non-zero payoff for the receiver) using the well-known solution concept of the

978-1-4799-3083-8/14/$31.00 ©2014 IEEE

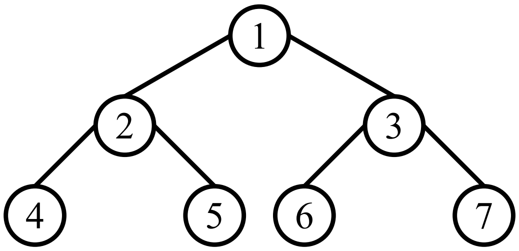


Fig. 1: A wireless network that consists of 7 nodes. Each node can send a packet to one of its 1 hop neighbors.

Nash Equilibrium (NE), *i.e.*, a strategy vector where no node has incentive in unilaterally deviating from its strategy. We discuss that a NE exists under any tree topology. We also show that, at a NE, an efficient scheduling of the transmissions is achieved, as there are no collisions, *i.e.*, a node either transmits successfully, or stays quiet. Moreover, we explain why finding a NE is different than simply aiming at finding a maximal transmission schedule, which is the standard goal of transmission scheduling algorithms [3] [4]. (iii) We present a distributed scheme that iteratively converges to a NE. We show that each NE is Pareto optimal. We study its efficiency, in the sense of speed of convergence to the NE and the number of successful transmissions at a NE through extensive simulations in different topologies.

II. System Model

We consider a single channel wireless network that consists of $N \geq 2$ wireless nodes, indexed by $\{1, 2, \ldots, N\}$. These nodes form an undirected graph $G = (V, E)$, where a vertex $v \in V$ corresponds to a wireless node and an edge $e \in E$ corresponds to a communication link that connects a pair of nodes (u, v). We consider tree topologies such as the one in Fig. 1. We denote the set of nodes that are 1 hop away from i with D_i and the number of nodes of this set with $|D_i|$.

We assume that time is divided in slots, nodes are synchronized in the sense that a transmission can happen only at the beginning of a slot, and that each packet needs exactly one slot to be transmitted. In addition, all queues are always full. As the proposed scheme converges fast enough, mobility does not need to be considered in this time-scale. We consider the unicast case, where a node i wants to send a packet to exactly one of its neighbors. Each node i has $|D_i| + 1$ options at each time slot: (i) To send a packet to a neighbor $d_i \in D_i$. We denote that option with T_{di}. (ii) To not transmit a packet (*i.e.*, to wait). We denote that option with W. We mention at this point that, when i transmits to d_i, all other 1 hop neighbors of i also receive the packet, but this packet is 'noise' for them, as it is not intended to them.

As a collision model, we assume that a collision occurs under the following circumstances (similar to the collision models typically assumed in the study of slotted Aloha [5]):(i) When a node receives packets simultaneously from at least two nodes, in which case all such packets collide. (ii) When node i transmits a packet to node j and node j also transmits. In this case, the transmission of i fails.

TABLE I: Payoff Models. Tx corresponds to transmission, Rx corresponds to reception.

Model #1		Model #2	
Status	Payoff	Status	Payoff
Successful Tx	*1-c*	Wait and Successful Rx	*1-e*
Wait	*0*	Successful Tx	*1-c*
Failed Tx	*-c*	Wait and No Rx	*0*
		Wait and Failed Rx	*-e*
		Failed Tx	*-c*

III. Graphical Game Model

To model the given graph setting as a non-cooperative game, we need to specify 3 elements: The players of the game and, for each player, his strategy, as well as his utility (payoff) function. Concerning the players, these are the N nodes of the graph that correspond to the wireless nodes. The strategy of a player i is one of the following: Either to transmit to one of his $|D_i|$ 1 hop neighbors, or to wait.

Concerning the payoff function of each player i, we should take into account the collision model of Section II. We study two payoff models that are summarized in Table I (the strategies are presented in decreasing order of payoff): Under Model #1, a receiver gets zero payoff no matter whether he receives successfully a packet or not. If a transmission is successful, a node receives a payoff $1 - c$, where 1 corresponds to the throughput from the transmission of the packet and $c \in (0, 1)$ is a constant that corresponds to the cost of transmission. If a transmission collides with another transmission, the payoff is just $0 - c = -c$. If a node chooses to wait, his payoff is $0 - 0 = 0$, as his throughput is zero and his cost of transmission is also zero. As a general rule, a successful transmission is preferable to waiting, which is also preferable to a failed transmission.

Under Model #2, the receiver can get a non-zero payoff too. We explicitly make the standard assumption that a node that transmits cannot receive, so we examine 3 cases for a node that waits: If he has a successful reception, he receives a payoff $1 - e$, where 1 corresponds to the net benefit from the reception of the packet and $e \in (0, c)$ is a constant that corresponds to the cost of decoding the packet. We point out that we assume that the decoding cost e is smaller than the cost of transmission c. If he cannot receive successfully a packet that is addressed to him, his payoff is $0 - e = -e$. If no packet is addressed to him, his payoff is $0 - 0 = 0$.

Depending on the application, payoff model #1 may be more preferable than payoff model #2 and vice versa. For example, if nodes are also interested in forwarding the packets that they receive, then payoff model #2 should be adopted.

For a general game with N players, in which each player has m possible strategies, the size of a normal form representation of the game would be $O(m^N)$, since the payoff of a player that chooses a particular strategy depends on his strategy and the strategy of the remaining $N-1$ players. Such a large representation would be needed if our network was fully connected. However, in multihop topologies the payoff of a player depends only on his strategy and the strategy of some of his neighbors. These games are called graphical games [6] and our model is an example of this type of game.

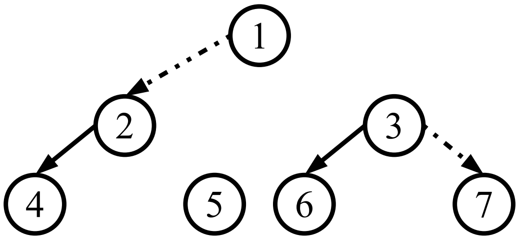


Fig. 2: Indicative NE for the network of Fig.1. The full arrows indicate the active transmissions at a NE under payoff model #1. The dashed arrows indicate the active transmissions at a NE under payoff model #2.

To identify the subset of neighbors that influence the payoff of a player i, we need to produce the square G^2 of the graph G, which is a graph that has the same set of nodes, but in which nodes i and j are neighbors when their distance in the graph G is at most 2 edges. In G^2, we compute the maximal degree d. If G is a tree and node i wants to transmit to node j, the payoff that he will receive depends on the strategy of j, as well as the strategy of all 1 hop neighbors of j, denoted by $|D_j|$. Consequently, the payoff is a function of $|D_j|+1$ nodes, *i.e.*, the number of the 1 hop neighbors of i in G^2. Therefore, the size of a graphical form representation would be $O(m^{|D_{\max}|+1})$, where $|D_{\max}|$ is the maximal degree. If $|D_{\max}| << N$ (which is the typical case), the size of the graphical representation of the game is much smaller than the one in a normal form game.

IV. Nash Equilibria

Having transformed this setup into an equivalent graphical game, we should address the fundamental question of the existence of a Nash Equilibrium (NE) in this game. As a first remark, we mention that, at any NE, the corresponding strategy vector $\mathbf{s}=(s_1,s_2,\ldots,s_N)$ should be collision-free. This is true since if a NE included collisions, then the nodes whose transmissions collided could improve their payoffs by simply deciding to wait. We then explain the difference of a NE under payoff model #1 from the notion of the maximal strategy (transmission) vector that plays a central role in transmission scheduling [3] [4]. Using similar terminology with [7], we call a strategy vector feasible if all nodes in the strategy vector either wait or have a successful transmission. A strategy vector is called a maximal strategy vector if adding an extra transmission will result in an infeasible strategy vector. All subsets of a maximal strategy vector are also feasible strategy vectors.

Though a NE under payoff model #1 fulfills the definition of a maximal strategy vector, a maximal strategy vector is not necessarily a NE under payoff model #1. To show that, consider Fig. 2. The strategy vector $\mathbf{s}=(s_1,s_2,s_3,s_4,s_5,s_6,s_7)=(T_2,W,T_7,W,W,W,W)$ is a maximal strategy vector since none of the nodes 2, 4, 5, 6 and 7 can have a successful transmission without interfering at least one active transmission. However, it is not a NE under payoff model #1 since node 2, being selfish, will transmit to either node 4 or node 5.

Under payoff model #2, it is easy to check that the above strategy vector is a NE. In general, the following properties hold (the proof is omitted): (i) All maximal strategy vectors are NE under payoff model #2 and (ii) Every NE under payoff model #1 is a NE under payoff model #2.

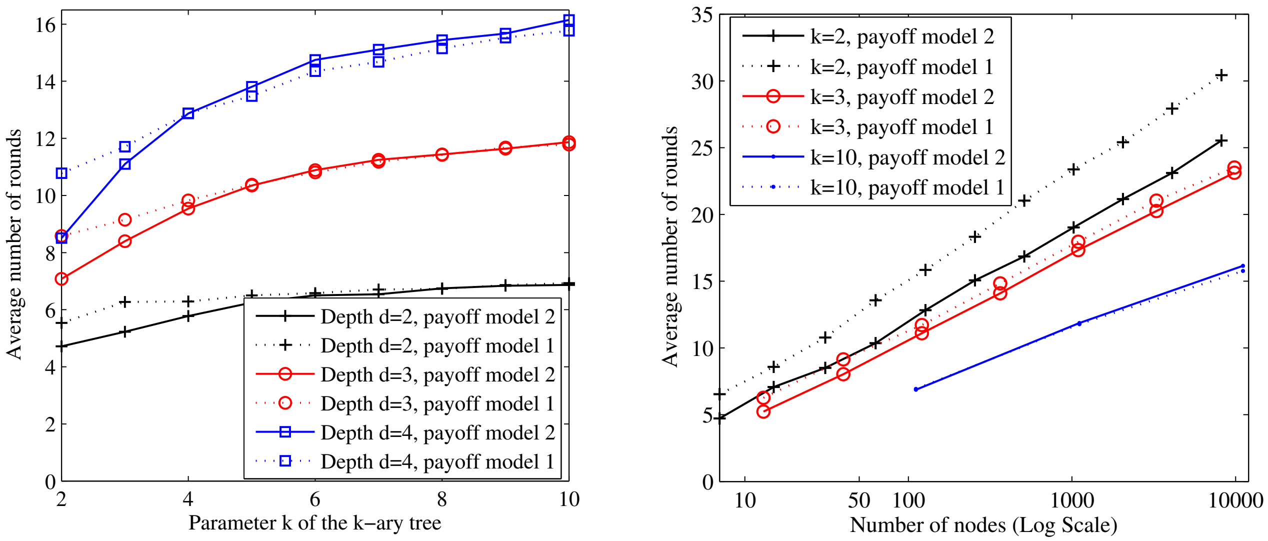
We now argue that, in this game, there is at least one NE, regardless of the payoff model used. Indeed, it is straightforward to construct a NE for each possible topology. For example, in Fig. 2 we have sketched a NE for a tree topology of 7 nodes under both payoff models. The corresponding strategy vector under payoff model #1 is $\mathbf{s}=(s_1,s_2,s_3,s_4,s_5,s_6,s_7)=(W,T_4,T_6,W,W,W,W)$ and the corresponding payoff vector is $\mathbf{u}=(0,1-c,1-c,0,0,0,0)$. We can check that, after reaching this strategy vector, no node can improve his utility on his own. The strategy vector under payoff model #2 is $\mathbf{s}=(s_1,s_2,s_3,s_4,s_5,s_6,s_7)=(T_2,W,T_7,W,W,W,W)$ and the corresponding payoff vector is $\mathbf{u}=(1-c,1-e,1-c,0,0,0,1-e)$. We can check that, after reaching this strategy vector, no node can improve his utility by simply changing his strategy on his own.

Moreover, a desired property of any NE of this game is that it is Pareto optimal. This is due to the fact that, under both payoff models, no node can improve his payoff at a NE without deteriorating the payoff of at least one node.

As a NE always exists, the question is how we can find it using a distributed scheme. A standard approach to find a NE is by applying the best response scheme [6]. In this scheme, each node chooses the strategy that, given the strategies of all other nodes, maximizes his payoff. Unfortunately, the best response scheme does not necessarily converge to a NE for this particular game as it may lead to oscillations. As a counterexample, let us consider a topology consisting of 2 nodes: {1-2}. It is straightforward to check that, at a NE, either node 1 will transmit to node 2 or vice versa. If both nodes choose as their initial strategy to wait, the best response strategy for each node is to transmit, which will lead to a collision. Then, the best response strategy for both of them will be to wait, afterwards the best response strategy for both of them will be to transmit, and so on. Therefore, the algorithm will never converge to one of the two NE of this game.

Next, we discuss a distributed iterative algorithm that aims at finding a NE under payoff model #1. The high level description of the algorithm is the following: Initially, each node has $|D_i|+1$ strategies, where D_i is the set of his 1 hop neighboring nodes. Each strategy is selected with a probability equal to $\frac{1}{|D_i|+1}$. Each strategy has the same probability since, under our model, a node is not interested in transmitting at a particular node, he simply wants to transmit a packet to any of his 1 hop neighboring nodes. Nodes select their strategies simultaneously. Then, assuming the existence of a collision-free feedback mechanism, d_i informs i whether his transmission is successful or not. Then, i computes his payoff on this round.

Each node i that has a successful transmission to node d_i transmits at the next round to the same node. This imposes some limitations on the strategies of the 1 hop neighbors of i and d_i for the next round. More specifically, (i) None of the 1 hop neighbors of i should transmit to i in the following round as no successful transmission can arise and (ii) None of the 1 hop neighbors of d_i (except, of course, i), that are



(a) Average round for convergence to a NE vs. parameter k and depth d.

(b) Average round for convergence to a NE vs. number of nodes.

Fig. 3: Performance evaluation: Average number of rounds for convergence to a NE as a function of (i) parameter k and depth d of the k-ary tree and (ii) the number of nodes.

also 2 hops neighbors of i, can transmit to d_i in the following round as no successful transmission can arise. The above piece of information is passed through the exchange of local 1 hop multicast messages that are sent by i and d_i correspondingly.

Each node that does not have a successful transmission takes into account these limitations to decide his strategy in the next round. Let V_i be the set of inactive neighboring nodes for node i, meaning that i cannot transmit successfully to any of them in the next round and $D_i - V_i$ be the set of active nodes. If $D_i - V_i = \oslash$, then node i should wait in the next round. Else, he chooses to wait with a probability equal to $\frac{|V_i|+1}{|D_i|+1}$ and to transmit to one of the active nodes with probability equal to $1 - \frac{|V_i|+1}{|D_i|+1} = \frac{|D_i|-|V_i|}{|D_i|+1}$. The motivation under this choice is that, as a node cannot transmit to nodes in $|V_i|$, the probability of nodes in V_i is transferred to the probability of waiting. As in the initialization phase, the probability to transmit to a particular node remains $\frac{1}{|D_i|+1}$. The algorithm ends when each node either has a successful transmission or waits and cannot have a successful transmission.

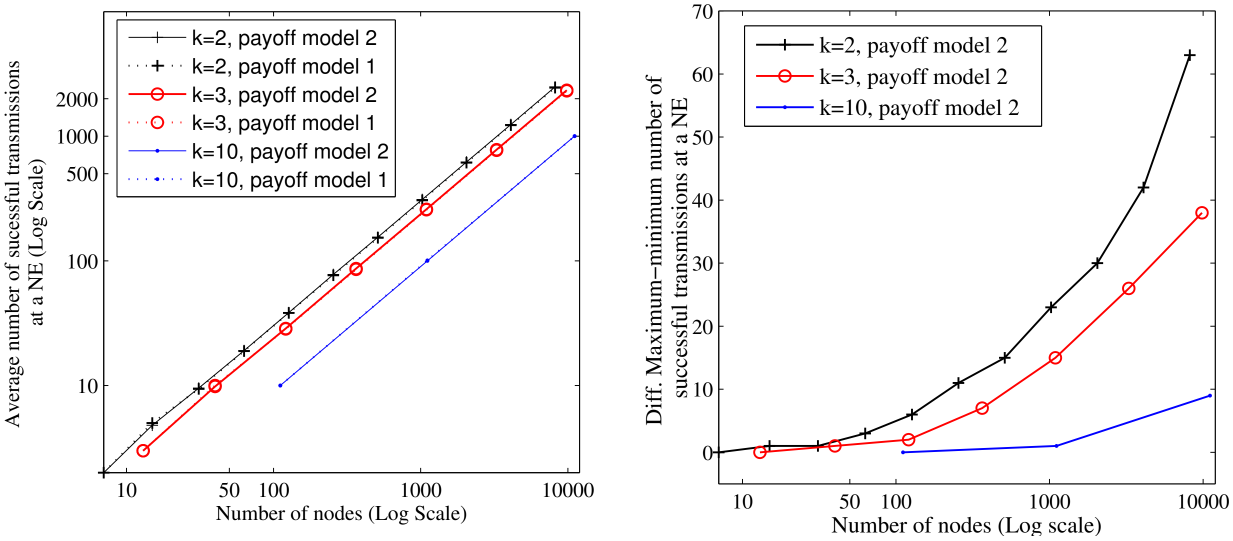
Under payoff model #2, there are only two differences in the algorithm: (i) Not only each node i that transmits successfully but also each node d_i that receives successfully will not change his strategy in the next round. This is because a successful reception leads to the biggest payoff under this payoff model as shown in Table 1. (ii) The algorithm ends when each node either has a successful transmission/reception or waits with no packet addressed to him and cannot have a successful transmission.

Due to the fact that the stopping condition of the algorithm corresponds to a strategy vector that is a NE, it is certain that, if the algorithm ends at a particular round, a NE will be reached under both payoff models. The other possibility is that the maximum number of iterations will be reached without a NE. However, as we show in the next section, the algorithm converges to a NE after a very modest number of iterations.

V. Performance Evaluation

We have simulated the proposed scheme to evaluate its performance when the topology is a perfect k-ary tree. Parameter k corresponds to the maximum number of 1 hop next level neighbors that each non-leaf node has. In our simulations, each non-leaf node has exactly k 1 hop next level neighbors. When this condition holds, all leaf nodes are on the same depth d and the k-ary tree is called perfect. For example, the tree in Fig. 1 is a perfect 2-ary tree. It is easy to show that a perfect k-ary tree of depth d has $\frac{k^{d+1}-1}{k-1}$ nodes. We simulated k-ary trees from few nodes up to more than 10,000 nodes and, for each k-ary tree, we have executed 1,000 simulations. The maximum number of rounds per simulation was set to 50 and the algorithm found a NE in all simulations for every setup.

The first set of simulations is used to evaluate the average number of rounds so that the algorithm converges to a NE versus the number of nodes/parameter k and depth d. Fig. 3a presents these results for trees of depth 2, 3, 4. We can see that the scheme converges fast to a NE demanding at most 16 iterations. The results are similar for both payoff models. We notice that, for a given parameter k, the number of rounds to converge to a NE is increasing with the depth d. This is natural since more nodes compete for spectrum access. However, the increase is quite slow. Moreover, for a given depth, the number of rounds slightly increases with parameter k. This is natural, since, again, more nodes compete for spectrum access. However, the effect of parameter k is smaller than the effect of parameter d, implying that the depth of the tree influences more the convergence speed of the algorithm than the density of the nodes in a particular level. We then present the average number of rounds to converge to a NE as a function of the number of nodes of trees with k=2, 3 and 10. In Fig. 3b the curves do not start/end from/to the same number of nodes, as we study trees of depth at least 2 (trees of depth 1 are trivial to be resolved) that contain at most ≈10,000 nodes. The average number of rounds to converge to a NE increases



(a) Average number of successful transmissions at a NE.

(b) Maximum/minimum number of successful transmissions at a NE.

Fig. 4: Analysis of the average/maximum/minimum number of successful transmissions at a NE as a function of the number of nodes in a k-ary tree.

almost linearly with the logarithm of the number of nodes and this is the reason that the convergence is very fast. This is true for both payoff models. NE under payoff model #1 demand margninally more rounds to converge, which is rather expected since more strategy vectors correspond to a NE under payoff model #2. It is interesting that trees with $k = 2$ need more rounds than trees with $k = 10$. This is justified as follows: Consider a tree of around 100 nodes. It can be constructed either as a 10-ary tree of depth 2 or as a 2-ary tree of depth 7. As we saw from Fig. 3a, the effect of the depth is bigger than the effect of the number of nodes in a particular level and this means that, for similar number of nodes, a longer tree demands more rounds to converge to a NE.

We then examine the average number of successful transmissions at a NE as a function of the number of nodes. We present these results in log-log scale in Fig. 4a. As expected, the number of successful transmissions increases linearly with the number of nodes. For both payoff models, the results almost coincide. Again, for similar number of nodes, we notice that the number of successful transmissions is bigger for longer trees (*i.e.*, nodes with smaller k). In Fig. 4b, we plot the difference: The NE with the maximum number of successful transmissions minus the NE with the minimum number of successful transmissions. We plot the results only under payoff model #2, as the results from payoff model #1 are very similar. The motivation is to examine whether there are NE that are (non-)preferable under this metric due to the fact that significant fewer/more transmissions take place. Indeed, as simulations show, any NE under a k-ary tree setup is almost equally preferable. For example, consider a 2-ary tree of depth 12 that has on average 2487 successful transmissions. The worst NE involves 2456 successful transmissions and the best NE involves 2519. The (plotted) difference of 63 corresponds to 2.5% fewer successful transmissions than the best possible case, which is insignificant.

VI. Related Work

Mackenzie et al. [8] were the first to propose the modeling of Slotted Aloha as a non-cooperative game and analyze the NE of the game. Payoff model #1 is inspired by this work. In [9], the authors relax the assumption that each node has a packet to send at each time slot. Moreover, they also consider a team optimization approach (though without applying coalitional game theory). In [10], the authors use pricing in the payoff function to motivate the nodes to cooperate. By enforcing cooperation through pricing mechanisms, the throughput of the centralized slotted Aloha can be achieved in a distributed network in which selfish users access the network attempting to maximize their own utility. However, in contrast to our work, these approaches consider a fully connected wireless network, where all nodes interfere with each other and consequently only one among the N nodes of the topology is able to transmit successfully at each slot.

In [11], the authors consider a single-cell wireless LAN providing a general game-theoretic framework for designing contention based medium access control protocols. Various utility functions are proposed and conditions for the existence and the uniqueness of a NE are derived. Simulation results show that the framework can achieve superior performance over the standard IEEE 802.11 Distributed Coordination Function (DCF). Again, the assumption is that every wireless node can hear every other node in the network.

Graphical games have already been applied a few times in wireless networks. In [12], the authors study channel selection for cognitive radio networks. Each secondary user chooses a channel to transmit assuming that only his neighboring nodes that have chosen the same channel cause non-negligible interference to him. The target is the minimization of the total regret. The no-regret approach is used in conjunction with other learning techniques to find a NE of the game.

In [13], the authors study the same problem using graphical games (even though they call them 'local interaction games') and propose two approaches: a) the minimization of the number of competing neighbors (aiming at network collision minimization) and b) an altruistic payoff that includes also the payoff of his neighbors (aiming at network throughput maximization). Contrary to our work, these communication targets correspond to nodes that belong to the same operator.

There is a substantial body of work on the topic of transmission scheduling in multihop wireless ad hoc networks. A work close to ours is the Five-Phase Reservation Protocol (FPRP) [3] that is used for distributed scheduling. Similar to our approach, the scheme is based on local interactions among the neighboring nodes that examine whether they can have a successful transmission (and inform their neighbors when they achieve it). However, contrary to our scheme, FPRP is used only for multicast transmissions and the target is simply to schedule the transmissions to find a maximal transmission vector and not to find a NE of the game. This is not necessarily a NE, as we already showed. A maximal transmission vector is not always a unanimously desirable outcome. Given the fact that nodes are selfish (this assumption holds in FPRP even if nodes do not follow a game-theoretic approach), FPRP may produce a strategy vector where there will be at least one node that could have a successful transmission and he is enforced to stay quiet. Finally, in some corner cases, the FPRP algorithm leads to a transmission vector where a node both transmits successfully and receives packets that cannot decode. In our approach, this will never happen at a NE. Another distributed scheduling algorithm that aims at eliminating collisions is presented in [4]. Each time slot is divided in six mini-slots and the first five of them are used by neighboring nodes that exchange control messages aiming at reserving the channel. If the channel is guaranteed to be idle, a transmission occurs. The approach considers both multicast and unicast transmissions. Simulations show that the performance of this scheme is similar to FPRP.

VII. Discussion & Work-in-Progress

We considered a multihop wireless network with a tree topology where selfish nodes compete for channel access and modeled it as a graphical game with two alternative payoff models. We presented a distributed scheme that leads to an efficient NE and evaluated its performance by simulation. We showed that the algorithm converges fast to a NE and, at a NE, the number of successful transmissions is almost equal.

One future direction is to study the multicast transmission scheme, where each node aims at sending his packet to all neighbors that are 1 hop away from him. Clearly, each node can choose between two strategies: To transmit (T) or to wait (W). Concerning the payoff, the transmission cost will be equally divided to the number of nodes to whom the packet is sent.

Another direction is to study this setup under repeated games, where a given game is played multiple times by the same set of players. The game that is repeated is called the stage game. Consider a linear topology that consists of 4 nodes: {A-B-C-D}. (W, T, T, W) is a strategy vector that corresponds to a NE for the multicast case of this stage game under payoff model #1. However, if there were two repetitions of the game, nodes B and C could make an agreement to transmit in different rounds. For example, in round 1, the strategy vector would be (W, T, W, W), whereas in round 2 the strategy vector would be *(W, W, T, W)*. Clearly, in both rounds, neither node A nor node D has motivation to change his strategy from W. Let us compare the payoff of nodes B and C if they simply decide to choose the strategy of the stage game in both rounds with the above mentioned scheme. Repeatedly applying the strategy of the stage game leads to a payoff of $2(1 - c) = 2 - 2c$. On the other hand, the proposed scheme leads to a payoff of $2 - c + 0 = 2 - c$, which is clearly preferable.

Acknowledgment

Vaggelis G. Douros is supported by the HERAKLEITOS II Programme which is co-financed by the European Social Fund and National Funds through the Greek Ministry of Education.

The research of Stavros Toumpis has been co-financed by the European Union (European Social Fund ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) Research Funding Program: THALES. Investing in knowledge society through the European Social Fund.

References

[1] K. Akkarajitsakul, E. Hossain, D. Niyato, and D. I. Kim, "Game theoretic approaches for multiple access in wireless networks: A survey," *IEEE Communications Surveys & Tutorials*, vol. 13, no. 3, pp. 372–395, 2011.

[2] K. Doppler, M. Rinne, C. Wijting, C. Ribeiro, and K. Hugl, "Device-to-device communication as an underlay to LTE-advanced networks," *IEEE Communications Magazine*, vol. 47, no. 12, pp. 42–49, 2009.

[3] C. Zhu and M. S. Corson, "A five-phase reservation protocol (FPRP) for mobile ad hoc networks," *Springer Wireless Networks*, vol. 7, no. 4, pp. 371–384, 2001.

[4] Z. Tang and J. Garcia-Luna-Aceves, "Collision-avoidance transmission scheduling for ad hoc networks," in *Proc. IEEE International Conference on Communications (ICC)*, 2000.

[5] D. P. Bertsekas, R. G. Gallager, and P. Humblet, *Data Networks*, 2nd ed. Prentice-Hall International, 1992.

[6] M. Kearns, *Graphical Games, chapter 7 of Algorithmic Game Theory*. Cambridge University Press, edited by N. Nisan et al., 2007.

[7] G. Lu and B. Krishnamachari, "Energy efficient joint scheduling and power control for wireless sensor networks," in *Proc. IEEE Conference on Sensor and Ad Hoc Communications and Networks (SECON)*, 2005.

[8] A. B. MacKenzie and S. B. Wicker, "Selfish users in Aloha: A game-theoretic approach," in *Proc. 54th IEEE Vehicular Technology Conference (VTC2001-Fall)*, 2001.

[9] E. Altman, R. El Azouzi, and T. Jiménez, "Slotted Aloha as a game with partial information," *Elsevier Computer Networks*, vol. 45, no. 6, pp. 701–713, 2004.

[10] D. Wang, C. Comaniciu, and U. Tureli, "Cooperation and fairness for slotted aloha," *Springer Wireless Personal Communications*, vol. 43, no. 1, pp. 13–27, 2007.

[11] T. Cui, L. Chen, and S. Low, "A game-theoretic framework for medium access control," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 7, pp. 1116–1127, 2008.

[12] H. Li and Z. Han, "Competitive spectrum access in cognitive radio networks: Graphical game and learning," in *Proc. IEEE Wireless Communications and Networking Conference (WCNC)*, 2010.

[13] Y. Xu, J. Wang, Q. Wu, A. Anpalagan, and Y.-D. Yao, "Opportunistic spectrum access in cognitive radio networks: Global optimization using local interaction games," *IEEE Journal of Selected Topics in Signal Processing*, vol. 6, no. 2, pp. 180–194, 2012.