A Novel Topology-blind Fair Medium Access Control for Wireless LAN and Ad Hoc Networks

Z. Y. Fang and B. Bensaou
Computer Science Department
Hong Kong University of Science and Technology
Clear Water Bay, Kowloon, Hong Kong
{zyfang—brahim}@cs.ust.hk

***Abstract*— This paper introduces a new backoff mechanism for IEEE802.11 which aims to achieve fair channel access without knowledge of the network topology. By adjusting a time interval and the contention window dynamically, the algorithm aims to approach the optimal equilibrium where the time interval is the minimum possible such that every node that faces the same contention successfully sends only one packet per such interval. We show how this algorithm can be modelled as a game and use game theoretic arguments to prove the existence and uniqueness of the equilibrium as well as convergence of the algorithm to this equilibrium.**

I. Introduction

Due to its great success both in infrastructure mode and in ad-hoc mode, wireless local area networking is being extensively investigated in the research and development communities. At the root of such success lies the efficiency of the standard medium access control scheme designed within, and adopted by, the IEEE 802.11 workgroup to serve as a distributed means of sharing a common broadcast channel between competing stations. At the basis of any access to the channel in IEEE 802.11 is the so-called distributed coordination function (DCF) which allows network nodes to compete for channel access by deploying a set of methods such as carrier sensing, random backoff, deference (or virtual backoff), and collision avoidance. Such methods are set in IEEE 802.11 such that the throughput of the network is maximised. More recently, with the adoption of IEEE 802.11 DCF as a de facto standard for simulating routing in mobile ad-hoc networks, a problem of fairness has been detected in IEEE 802.11. In short, when there are N network nodes competing with each other, the backoff algorithm adopted by IEEE 802.11 tries to adjust the channel access attempt probability of each node such that it approaches the optimal (theoretical) success probability of $1/N$, within the shortest possible time interval, thus effectively achieving very high throughput. In the context of wireless LAN where all stations are within radio reach of each other this method achieves its goals perfectly. However, when the network topology is not symmetric, and there are terminals that are hidden from each other such goals are not achieved by the adopted technique, and some stations are at a disadvantage when accessing the channel. This unfairness can be very marked when the traffic load is high.

Many schemes have been proposed in the literature to overcome this problem. As discussed in [1], these schemes can be categorised into two types: centralised and distributed. Among the benefits of the distributed schemes, are their tolerance to both node and link failures, their reduced implementation overheads and their suitability for battery powered devices. In addition, distributed schemes are more applicable for mobile ad-hoc networks (MANETs) whose prominent features are their dynamic topology and infrastructureless architecture. From another point of view, the proposed schemes can also be classified as either scheduled, or contention-based or a combination of the two, as suggested in [2]. Scheduled schemes generally need coordinators to dictate the behaviour of nodes, thus are generally centralised, otherwise, when they are distributed they require a large amount of signalling to coordinate the nodes. However contention-based schemes are more likely to be distributed.

When designing a fair protocol, the objective of fairness is the first issue to be addressed before considering how to maintain such fairness. Many such objectives have been discussed in the literature. Among them, the most popular ones are max-min fairness [3] whose goal is to provide a minimum bandwidth guarantee to every flow while maximising the total throughput of the network and proportional fairness [4] whose goal is to assign bandwidth to flows proportional to their requirements. As discussed in [1], bandwidth can be assigned to competing flows or nodes according to predefined weights similar to the traditional weighted fair queueing in wired networks. In this spirit, a distributed fair scheduling algorithm which is derived from IEEE 802.11 DCF is presented in [1]. This algorithm tries to provide bandwidth guarantees to the flows in proportion to their weights by choosing the appropriate backoff timers. The algorithm needs to incorporate a weight allocation method as well as the means to distribute such information among nodes. A measurement-based fair backoff algorithm is introduced in [5–7]. In cooperation with the max-min fair share allocation algorithm proposed in [8], this algorithm can achieve max-min fairness and with other fair share assignment techniques can achieve generally weighted-proportional fairness. While these algorithms, as well as other scheduling algorithms [2], can alleviate the fairness problem that occurs in IEEE 802.11, they however need to exchange information between nodes (flows) either to obtain the fair (weights) allocation information or to maintain topology knowledge in order to achieve fairness. Considering a given pricing scheme, by maximising a specific utility function, the algorithm proposed in [9] implements successfully a topology blind backoff scheme to provide proportional fairness.

In this paper, following the same principle as in [9], we propose a novel backoff algorithm that alleviates the problem of fairness encountered in IEEE 802.11 without need for any exchange of information between the nodes nor any assumption about any global pricing scheme in the network. The algorithm is self adapting to changes in the traffic load as well as topological changes in the network. A game theoretic model of the algorithm shows that the algorithm admits an equilibrium and thus is intrinsically stable.

The rest of this paper is organised as follows. Section II reviews the basic functions provided by IEEE 802.11 DCF. A novel window adjustment technique is introduced in Section III. The convergence and stability properties of the algorithm are analysed using a game theoretic approach in Section IV. A simulation scenario is discussed in Section V. Finally, we draw our conclusions in Section VI.

II. IEEE802.11 DCF

The basic medium access protocol in IEEE802.11 is the distributed coordination function (DCF) that allows for automatic medium sharing between stations through the use of CSMA/CA, a set of carefully chosen inter-frame space times (IFS) and a random backoff time following a busy medium condition. In addition, all directed traffic uses immediate positive acknowledgement (ACK frames) and retransmissions are rescheduled by the sender if no ACK is received. The CSMA/CA protocol is designed to reduce the collision probability between stations at the point where collisions would most likely occur. For this purpose, DCF mandates that each station defers for a time equal to DIFS and if the medium is still idle a random backoff procedure is invoked to resolve medium contention conflicts. In IEEE 802.11, carrier sensing is performed both through physical means as well as a virtual mechanism. Physical carrier sensing is done by the physical layer while virtual carrier sensing is performed by the MAC layer: each station that sends a RTS, CTS or DATA packet sets a time field in the header that indicates the time it takes the ongoing transmission to be completed. Any station that overhears either packet, updates its so-called network allocation vector (NAV) which mandates how long the station will defer before trying to compete for the channel again.

DCF defines two types of access methods, a basic access method in which a station immediately sends data packets after the backoff. This method is usually invoked for packets that are shorter than a predefined threshold (RTSThreshold), otherwise, for long packets, IEEE 802.11 uses the RTS/CTS method which uses control packets to distribute timing information to other stations and thus avoids collisions during the transmission of the long data packet. This ensures a high network throughput. In the RTS/CTS method, upon completion of the backoff procedure, the node sends a RTS packet to which the receiver replies with a CTS packet after a short IFS (SIFS). The two packets have the effect of inhibiting all nodes within radio transmission range of the sender and all nodes within radio transmission range of the receiver, for the whole period of time indicated by the timing information in the packets' headers. This almost guarantees that no collision will occur in the following period of time.

For hardware efficiency, DCF uses a discrete backoff timer thus requiring only integer logic. The binary exponential backoff algorithm is adopted in order to adapt the backoff timers to the traffic density as fast as possible. At each transmission, the backoff timer is chosen uniformly from the range $(0, w-1]$, where w is the node's contention window. Initially (respectively, upon each successful transmission) w is set (respectively reset) to CW_{min} (usually 32). Upon each unsuccessful transmission[1], the contention window w is doubled until the maximal value CW_{max} (usually 1024).

In networks where node density and thus competition is inhomogeneous IEEE 802.11 suffers from unfairness because of the BEB. In such situations some nodes will always succeed because they are hidden from their competitors and will reset their contention window to CW_{min} most of the time, while others who can overhear the activity in their vicinity will often double their contention window (see for example [6]). This leads to some nodes enjoying statistically shorter backoff timers than others and thus to the former enjoying higher throughput than the latter.

Analysing carefully such a problem, one realises that it is mainly due to the fact that while the backoff procedure of IEEE 802.11 guarantees that ultimately a node will have a chance of attempting to access the channel (since the backoff timers are frozen when a node defers and decrease when contending for the channel), it does not guarantee the success of such attempts (in presence of hidden terminals). This is particularly more dramatic when the traffic load is high and the node density is non uniform (i.e., when stations are greedy and the station under consideration competes with many others who do not see the same amount of competition): the probability of a station succeeding a large number of transmission consecutively in an asymmetric network is non negligible and thus the probability of another station doubling its contention window, is thus equally non negligible.

III. Improving the fairness of the backoff algorithm

In the proposed algorithm, the main goal is to give stations that face the same level of competition the same probability of successfully transmitting one packet within an optimal (shortest possible) period of time, irregardless of the nature of the contention. In other words if two stations face the same amount of competition irrespective of whether one of them is hidden from one or more of its competitors than the other, the two stations should achieve the same throughput. That is,

[1] In fact IEEE 802.11 mandates that a station retry counter is incremented at each unsuccessful transmission and that when it reaches a given threshold, the contention window is doubled. However, the standard does not specify a value for such a threshold. NS-2 implementation of 802.11 sets this threshold to 4 for long packets and 7 retries for short ones.

assuming that in a given neighbourhood, N nodes compete with each other, ideally our algorithm tries to approach the equilibrium point where each one of them has the same probability $(1/N)$ of successfully transmitting one packet within an optimal period of time t_p equal to $N \times t_{unit}$, where t_{unit} is the time it takes to succeed a four way handshake RTS-CTS-DATA-ACK. The major hurdle to achieving this is that in the presence of hidden terminals, nodes do not know the topology of the network and thus do not know the extent of the competition, as failure of transmission can occur due to collisions either at the sender or at the receiver while the backoff procedure takes place at the sender only. Therefore it becomes one major task of the algorithm to choose/estimate dynamically the period of time t_p. Essentially each sender counts the number of successful transmissions in a period of time and either increases the contention window to relinquish bandwidth to others in the case of disproportionate successes or decreases it in the case of successive failures to become more aggressive (this is fundamentally different from the BEB). Once the contention window hits either the maximum or the minimum bound, with persistence of the disproportionate successes or failures, the estimate of the time interval is obviously non optimal (either too large or too small) and thus it is either reduced or increased respectively.

The pseudo-code for this is given in Algorithm 1 which is represented as a set of interrupt handlers that are triggered by the corresponding events. Namely, Initialisation event (when the protocol is started), packet transmission event, and timer event (which is triggered each t_p^i units of time). States, define the state variables of the algorithm. Contention window win_i and time interval t_p^i are two parameters that are used by node i to adjust its channel access probability and both parameters are adjusted based on the node's success probability. This addresses the problem as identified in IEEE 802.11. Intuitively, one can see that t_p^i for node i increases when the number of contenders increases. Since each node i initialises t_p^i to one packet time and the contention window to the minimum, statistically all nodes have a much higher probability of increasing the time interval first (since none but one of them can succeed at most once within such a short t_p^i. After increasing t_p^i to some extent, nodes will start succeeding more than once within such large intervals and thus start relinquishing their "turns" by increasing their contention window. Since the adjustment of the time interval occurs in a much larger time scale than that of the contention window, once the optimal value of t_p^i is reached (at equilibrium), the contention window will still oscillate within a small interval about the optimal value due to the statistical nature of the channel access.

Algorithm 1 Distributed Fair MAC

```
States:
  t^i_packet, current packet length.
  t^i_unit, 4 way handshake duration. This is already available in 802.11
        in the outgoing NAV calculation
  k_i, the time interval in terms t^i_unit.
  t^i_p, current frame time interval as seen by node i.
  n_i, the number of successful transmission.
  win_i, contention window.
Events:
 Init:
    t^i_packet = 0.
    k_i = 1.
    t^i_unit = t_RTS + t_CTS + t_ACK + 3 * t_SIFS + t_DIFS + t^i_packet.
    t^i_p = k_i * t^i_unit.
    Arm timer TP with time t^i_p.
 Channel Access:
    Use current contention window win_i to access channel.
    If ( successfully received ACK packet )
        n_i = n_i + 1.
        Update t^i_packet.
        Update t^i_unit.
 Timer Event: (upon expiry of TP)
    if (n_i > 1)
        if ( win_i == CW_max )
            k_i = k_i - n_i + 1; t^i_p = k_i * t^i_unit
        else win_i = 2 * win_i ;
    else
        if ( n_i == 0 )
            if ( win_i == CW_min )
                k_i = k_i + 1; t^i_p = k_i * t^i_unit
            else win_i = win_i/2 ;
        endif
    endif
    n_i = 0.
    t^i_p = k_i * t^i_unit.
    Arm a new timer event TP with time t^i_p.
```

IV. Stability and Convergence of the Algorithm: A Game Theoretic Model

In MANET, channel resources are allocated in terms of link flows. A flow contention graph that is constructed based on the node graph can be used to capture the contention relations among flows. The vertex in a flow contention graph represents an active link flow, and the edge between two vertices denotes the conflict between the two flows. Each maximal clique (a complete sub-graph) in the flow contention graph represents a "channel resource". Denote the set of flows as $\mathcal{N} = \{1, \cdots, N\}$. The transmission rate for flow i is defined as $x_i, i = 1, \cdots, N$. The set of maximal cliques in the flow contention graph is denoted as $\mathcal{M}= \{1, \cdots, M\}$. Each clique in $\mathcal{M}$ has a capacity $c_j, j = 1, \cdots, M$ (notice that in a general conflict graph such capacities cannot be normalised). One flow may belong to several maximal cliques in the flow contention graph. These relations between flows and cliques can be described by a bi-partite graph described by an $M \times N$ matrix A as follows:

$$a_{j,i} = \begin{cases} 1, & \text{if flow } i \text{ belongs to clique } j, \quad i \in \mathcal{N} \\ 0, & \text{if flow } i \text{ does not belong to clique } j, \quad j \in \mathcal{M} \end{cases}$$

The capacity constraints can thus be written in the following matrix form as:

$$\boldsymbol{Ax} \leq \boldsymbol{C} \tag{1}$$

where $\boldsymbol{x} = (x_1, \cdots, x_N)^t$ is the flow rate vector and $\boldsymbol{C} = (c_1, \cdots, c_M)^t$ is the clique capacity vector. In addition, each flow should obtain a positive flow rate, thus:

$$x_i > 0 \qquad i = 1, \cdots, N \tag{2}$$

Under the control of the FMAC algorithm, flow i tries to transmit exactly one packet in a time interval t_p^i, determined by the integer k_i. This implies that flow i aims to minimise the value $(x_i \times k_i - 1)^2$. Every sender in a flow adjusts its flow rate in a selfish way (without knowledge of the behaviour of others) to reach this objective. The behaviour of the flows can thus be modelled as a non-cooperative game, which is called here the FMAC game, in which the set of active flows are the players. The strategy for player i is the tuple (x_i, k_i), where x_i satisfies conditions (1) and (2). The set of flow rate vectors Ω that satisfies (1) and (2) is called a feasible set. It can be easily shown that Ω is a nonempty, convex and compact set. The range for the time interval is the set of natural numbers $\mathbb{N}$. To simplify the analysis, we define a composite argument $\xi_i = x_i \times k_i$, ξ_i represents the average number of successful packet transmissions per interval of time. It follows that the range for ξ_i is the set of positive real numbers $\mathbb{R}^+$. Therefore the strategy space for flow i in terms of ξ_i is $U^i = \{\xi_i | \xi_i \in \mathbb{R}^+\}$, and the strategy space for the game is $U = \{(u_1, \cdots, u_n) | u_i \in U^i\}$. Obviously U is a superset of Ω. If the number of neighbour flows is bounded say by B (this is true in real networks), the minimal flow rate will be statistically bounded as we can show that the value of k_i is also statistically bounded (i.e., the probability that $k_i >> B$ is negligibly small). The strategy space can thus be truncated and hence U is also a convex and compact set. According to the players' incentive, the payoff function can be defined as $\phi_i = -(\xi_i - 1)^2$, which is a mapping from $U^1 \times U^2 \times \cdots U^N$ to $\mathbb{R}$.

A. Existence and Uniqueness of Nash Equilibrium

A non-cooperative game settles at a so-called Nash equilibrium if one exists. In the FMAC game, a vector of strategy $u^* \in U$ is called a Nash equilibrium if no player can increase its payoff by adjusting its strategy unilaterally, mathematically:

$$\phi_i(u^*) \geq \phi_i(u^*_{-i}, u_i) \qquad \forall u_i \in U^i, i = 1, \cdots, N \tag{3}$$

where u^*_{-i} denotes the strategies of all the other players other than i. For the FMAC game, we have:

Theorem 4.1: The FMAC game admits a unique Nash equilibrium in its pure strategies.

Proof: Let's consider the payoff function $\phi_i : U^1 \times \cdots \times U^N \to \mathbb{R}$. It is jointly continuous in all its arguments and strictly concave in u^i for every $u^j \in U^j, j \in \mathcal{N}, j \neq i$, and we have,

$$\phi_i(u^i, u_{-i}) \to \infty \quad as \quad |u^i| \to \infty, \quad \forall u_{-i} \in U_{-i}, \;\; i \in \mathcal{N}$$

According to Corollary 4.2 in [10], these conditions are sufficient to insure the *existence* of a Nash equilibrium for the FMAC game in its pure strategies.

Define now a weighted nonnegative sum of the function $\phi_i(\xi)$ as

$$\sigma(\xi, r) = \sum_{i=1}^{N} r_i \phi_i(\xi), \quad r_i \geq 0, \tag{4}$$

where $\xi = (\xi_1, \cdots, \xi_N)$. The pseudo-gradient of $\sigma(\xi, r)$ can be calculated as:

$$g(\xi, r) = \begin{bmatrix} r_1 \bigtriangledown \phi_1(\xi) \\ r_2 \bigtriangledown \phi_2(\xi) \\ \vdots \\ r_N \bigtriangledown \phi_N(\xi) \end{bmatrix},$$

and its Jacobian with respect to ξ can be computed as:

$$G(\xi, r) = diag(-2r_1, \cdots, -2r_N)$$

It follows that

$$G(\xi, r) + G^t(\xi, r) = diag(-4r_1, \cdots, -4r_N) \tag{5}$$

Equation (5) shows that for any positive vector r, $G(\xi, r) + G^t(\xi, r)$ is negative definite. Thus according to Theorem 6 in [11], $\sigma(\xi, r)$ is so-proved to be diagonally strictly concave. Therefore, according to Theorem 2 in [11], the property of $\sigma(\xi, r)$ together with the property of the strategy space U (convex compact set) are sufficient to insure that the equilibrium point of the FMAC game with respect to ξ is *unique*. ■

B. Convergence to Nash Equilibrium

We have shown that there exists a unique Nash equilibrium in the FMAC game. Now we can show that the FMAC algorithm leads the system to this equilibrium point. Based on the payoff function, we have $\frac{\partial \phi_i}{\partial \xi_i} = -2(\xi_i - 1)$. According to the FMAC algorithm, if $\xi_i > 1$, then $\frac{\partial \phi_i}{\partial \xi_i} < 0$, ξ_i decreases. If $\xi_i < 1$, then $\frac{\partial \phi_i}{\partial \xi_i} > 0$, ξ_i increases, thus we can use the following differential equation to describe the dynamics of the algorithm:

$$\frac{d\xi_i}{dt} = \dot{\xi}_i = r_i \frac{\partial \phi_i(\xi)}{\partial \xi_i} \quad i = 1, \cdots, N.$$

It can also be written as:

$$\dot{\xi} = g(\xi, r). \tag{6}$$

Having shown previously that $G + G^t$ is a negative definite quadratic form, according to Theorem 8 in [11], the system described by (6) is globally stable, which means the FMAC algorithm will lead the network to the unique Nash equilibrium.

C. Discrete Implementation of the FMAC Game and its Fairness Property

As we can see, the proposed FMAC algorithm is a discrete implementation of the FMAC game. In the FMAC algorithm, for flow i, the time interval is determined by the corresponding integer number $k_i \in \mathbb{N}^+$, which is a natural number. The transmission rate x_i is controlled by the contention window. Notice that the equilibrium is unique in terms of ξ_i, which is a product of x_i and k_i. Thus in the equilibrium, the contention window may not be unique. For example, if N flows form a complete flow contention graph (a wireless LAN configuration), then if all flows choose the same contention window, they will have the same flow rates. It turns out in

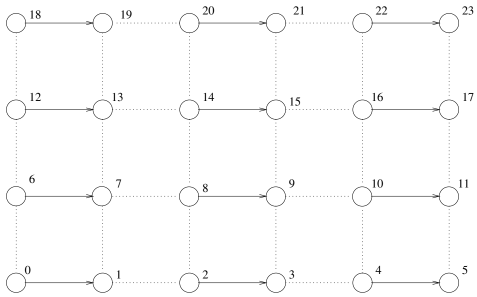

Fig. 1. Network topology

this case, every flow will choose the same integer $k = N$ (or slightly larger), and the payoff for every flow is 0. The corresponding strategy is the unique equilibrium, and the fairness is achieved in this point.

Consider the payoff function $\phi_i = -(x_i \times k_i - 1)^2$, the objective of the FMAC algorithm is to maximise ϕ_i. Since the objective function is a concave function, as shown in [12], maximising a concave utility function is equivalent to achieving some system-wide notion of fairness. Therefore the FMAC algorithm will achieve system-wide global fairness. This was proved in the scenarios with complete flow contention graph. In the scenarios with general contention graphs, it will be demonstrated by simulations.

V. Simulations

In order to evaluate the performance of the algorithm, we have replaced the BEB by the FMAC algorithm in the NS-2 implementation of IEEE 802.11, and did simulations with several scenarios that led to similar conclusions which we summarise here. In most simulations the algorithm proved to provide load proportional fairness irregardless of location (i.e., nodes that face the same amount of competition, despite being non competing nodes, achieve the same throughput). One major drawback to the algorithm (in fact we believe it is a "drawback" of fairness itself) is that the total network throughput achieved is lower most of the time than that achieved by the original IEEE 802.11/BEB. This perhaps makes the case for max-min fairness and perhaps the need for some local information exchange between sender and receiver [2], [8].

We examine here a network comprising 24 nodes (Fig. 1) out of which, each pair acts as a sender-receiver of a link (flow). Active links are represented by plain lines while dashed lines represent contention between nodes. The links are positioned on a grid such that the levels of competition are different for different nodes. In the sequel each link will be identified by its sender. The four links at the four corners of the grid, 0, 4, 18, and 22, face the same level of competition: each of them competes with two other links. The six median links on the sides of the grid (2, 6, 10, 12, 16, and 20) face a medium level of competition: each competes with 3 other links. Finally the two central links 8 and 14 face a high level of competition: each competes with 4 links. Moreover in this scenario links that face the same level of competition do not necessarily

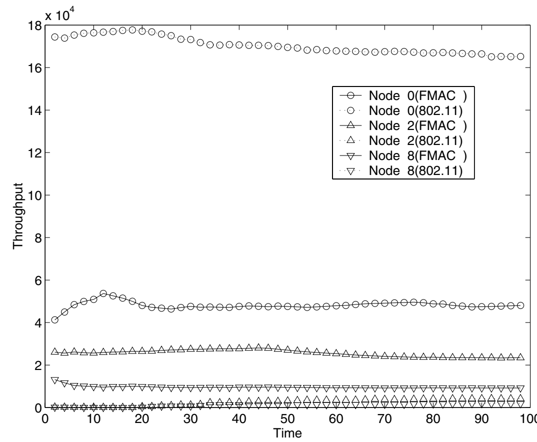

Fig. 2. Link goodput

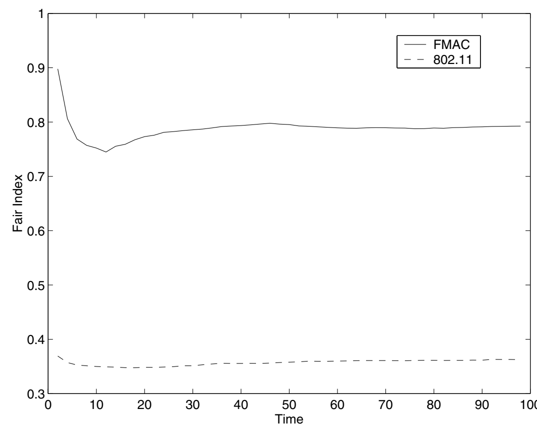

Fig. 3. Fairness index

compete with each other. Due to this discrepancy in the competition, the original IEEE802.11 (with the BEB) favours links 0, 4, 18 and 22, who obtain an extremely high share of the total throughput, while other links are penalised drastically. Our algorithm provides better fairness as can be seen in Fig. 2. In this figure, due to symmetry, we represent only links 0, 2 and 8, dashed lines correspond to the BEB while plain lines represent the FMAC algorithm. This latter guarantees nodes 0, 2 and 8, and their equivalent nodes, goodputs of 50Kbps, 25Kbps and 10Kbps respectively. IEEE802.11 on the other hand provides nodes 0, 2 and 8 with goodputs 160Kbps, 5Kbps and 1.5Kbps which is too dramatically unfair. This is confirmed by Fig. 3 where we show the fairness index computed as proposed in [13]. The closer the value of such index to 1 the better the fairness.

Finally, in the last three figures, we show the total throughput achieved by both protocols. It has recently become increasingly clearer [6], [7], [14] that fairness has a price and that most realistic protocols (which do not assume too many practically infeasible conditions) who try to improve fairness of IEEE 802.11 would have to pay a price for this in terms of total network throughput degradation. The questions that

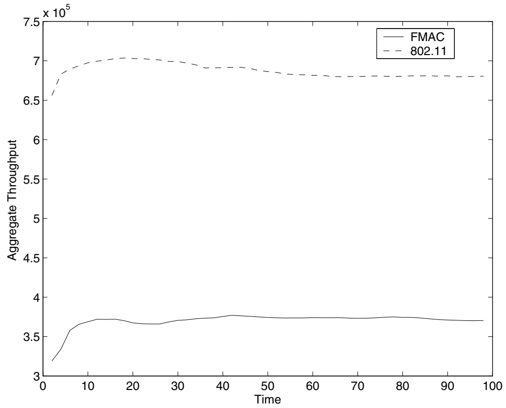

Fig. 4. Aggregate goodput:all flows

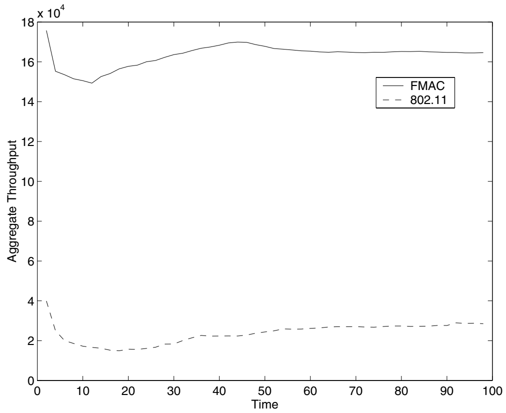

Fig. 5. Aggregate goodput: excluding links 0, 4, 18 and 22

remain are how much is acceptable and what is the trade-off. To answer these partly, the figures show three pairs of curves, one that compares the total goodput calculated over all transmitting nodes in Fig. 4, the second pair in Fig. 5 compares the total goodput excluding the corner nodes (0, 4, 18 and 20) and finally the last pair in Fig. 6 compares the aggregate goodput of central nodes (8 and 14) only. It is clear that the price paid is a dramatic total goodput degradation. Most of the time less than 50% throughput is lost due to introduction of fairness. However, nodes that are usually penalized unfairly in IEEE 802.11 achieve in the FMAC a much higher throughput, in fact such goodput is multiplied by a factor of 4 to 5. As a rule of thumb, should user satisfaction be unimportant, such degradation in the throughput (slightly better than 50% of IEEE 802.11 in this example, which is the worse we encountered) would be unjustifiable. However since user satisfaction is the driving force behind any fairness oriented work, such loss of total throughput is compensated by the possibility of providing guarantees irrespective of the network topology.

VI. Conclusion

A distributed fair MAC algorithm has been proposed in this paper. The key idea of the algorithm is to let nodes estimate the

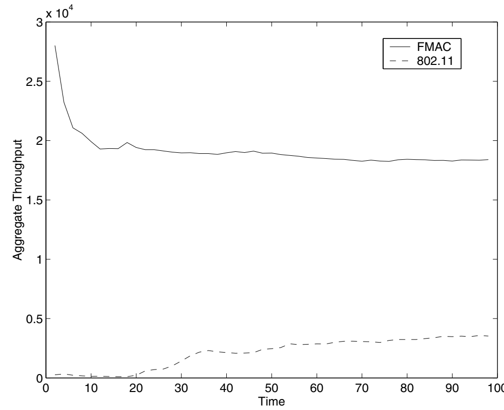

Fig. 6. Aggregate goodput: nodes 8 and 14

level of competition for bandwidth in their neighbourhood, and to ensure they relinquish channel access attempts whenever they achieve their estimated instantaneous fair throughput. To show the well fondness of the algorithm and prove stability of the algorithm irregardless of its topology-blindness, the protocol has been modelled as a dynamic non-cooperative game and the existence and uniqueness of an equilibrium has been shown. Simulations also have shown the convergence and fairness of the algorithm and highlight the price that one has to pay for fairness in wireless ad-hoc networks.

References

[1] N. H. Vaidya, P. Bahl, and S. Gupta, "Distributed fair scheduling in a wireless LAN," in *ACM MOBICOM*, pp. 167–178, 2000.

[2] L. Bao and J. J. Garcia-Luna-Aceces, "A New Approach to Channel Access Scheduling for Ad Hoc Networks," in *ACM MOBICOM*, pp. 210–221, 2001.

[3] D. Bertsekas and R. Gallager, *Data Networks*. Prentice Hall Inc., 1992.

[4] F. Kelly, "Charging and rate control for elastic traffic," *European Transactions on Telecommunications*, vol. 8, pp. 33–37, January 1997.

[5] B. Bensaou, Y. Wang, and C. C. Ko, "Fair Media Access in 802.11 based Wireless Ad-Hoc Networks," in *ACM MobiHOC*, pp. 99–106, 2000.

[6] Y. Wang and B. Bensaou, "Achieving Fairness in IEEE 802.11 DFWMAC with Variable Packet Lengths," in *IEEE Globecom'01*, pp. 3588–3593, 2001.

[7] Z. Y. Fang, B. Bensaou, and Y. Wang, "Performance Evaluation of a Fair Backoff Algorithm for IEEE802.11 DFWMAC," in *ACM MobiHOC*, pp. 48–57, 2002.

[8] X. L. Huang and B. Bensaou, "On Max-Min Fairness and Scheduling in Wireless Ad-Hoc Networks: Analytical Framework and Implementation," in *ACM MobiHOC*, pp. 221–231, 2001.

[9] T. Nandagopal, T. E. Kim, X. Gao, and V. Bharghavan, "Achieving MAC Layer Fairness in Wireless Packet Networks," in *ACM MOBICOM*, pp. 87–98, 2000.

[10] T. Basar and G. J. Olsder, *Dynamic Noncooperative Game Theory*. Acsdemic Press, Harcort Brace Company, 1995. Second Edition.

[11] J. B. Rosen, "Existence and uniqueness of equilibrium points for concave n-person games," *Econometrica*, vol. 33, pp. 520–534, Jul. 1965.

[12] S. Shenker, "Fundamental design issues for the future internet," *IEEE Journal on Selected Areas in Communications*, vol. 13, pp. 1176–1188, 1995.

[13] R. Jain, *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling*. New York, NY: Wiley-Interscience, April 1991.

[14] X. Yang and N. H. Vaidya, "Priority scheduling in wireless ad hoc networks," in *ACM MobiHoc*, pp. 71–79, 2002.