Throughput Modelling and Fairness Issues In CSMA/CA Based Ad-Hoc Networks

Xin Wang and Koushik Kar

Abstract— **In this paper, we consider the throughput modelling and fairness provisioning in CSMA/CA based ad-hoc networks. The main contributions are: Firstly, a throughput model based on Markovian analysis is proposed for the CSMA/CA network with a general topology. Simulation investigations are presented to verify its performance. Secondly, fairness issues in CSMA/CA networks are discussed based on the throughput model. The origin of unfairness is explained and the trade-off between throughput and fairness is illustrated. Thirdly, throughput approximations based on local topology information are proposed and their performances are investigated. Fourthly, three different fairness metrics are presented and their distributed implementations, based on the throughput approximation, are proposed.**

I. Introduction

As a broadcasting channel is shared by all the nodes in a wireless network and it is a limited resource, MAC (Medium Access Control) algorithms are essential to reduce collisions and ensure high system throughput. Apart from reducing bandwidth loss due to collisions, the MAC strategy should also ensure that the available bandwidth is shared by the competing nodes in some fair manner. Various distributed MAC protocols have been proposed in the literature to enhance system performance in the wireless environment. Among them is the CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) protocol, which is the basis for DCF (distributed coordination function) adopted in 802.11 MAC specifications [1] and hence has special importance.

In this paper, we address the throughput performance analysis and fairness provisioning issues in CSMA/CA networks. Although these two topics have been considered before (see [4], [5], [6], [9] and the references therein), these works suffer from several limitations. In [5], Bianchi proposed a simple model that accounts for all the exponential backoff protocol details and allows computation of saturation throughput of IEEE 802.11 DCF [1]. Although simulations in [5] demonstrated high accuracy, the model did not consider the network topology and could not characterize the collisions that result from the other active nodes in the neighborhood of the transmitter. In [6], [7], [8], and [9], the researchers focused on the fairness issues in wireless networks. However, the strategies proposed in these works are heuristic approaches, and a formal analysis of the fairness properties of the approaches is lacking.

Our work on CSMA/CA network throughput modelling uses the techniques similar to those used by Boorstyn et al. in [2]; however it differs significantly from the latter. Firstly, the authors in [2] have only analyzed throughput performance of a pure CSMA network which is not directly applicable to most MAC protocols of today. We model the throughput of the CSMA/CA network using the same line of analysis, but we take into account the RTS/CTS exchange in the CSMA/CA network and our work is readily extended to the throughput analysis of the IEEE 802.11 DCF. Secondly, the authors in [2] do not address any fairness issues. In contrast, we discuss fairness issues in CSMA/CA networks in detail, and present distributed algorithms, based on local topology information, that achieve fairness amongst the links in the CSMA/CA network.

A prominent feature of the wireless network is that its feasible rate region depends on the protocol, therefore the fairness problem needs to be considered in the context of a particular protocol. In this paper, we consider the throughput modelling and fairness issues in the CSMA/CA based ad hoc networks. We propose a throughput model for the CSMA/CA network with general topology, and investigate the performance of the model with simulations. Based on the model, we can well explain that unfairness originates from the "hidden node problem" and the trade-off between throughput and fairness. Although our throughput model needs topology information of the entire network, we show how the throughput can be approximated with only local topology information. The performance of the approximation is analyzed and investigated with simulations. Based on the throughput approximation, we discuss the fairness provisioning issue in the CSMA/CA networks, and propose three distributed scheduling methods to provide fairness among the contending users.

This paper is organized as follows. In Section 2, after a brief overview of the CSMA/CA network, we present the system model and the assumptions. The throughput model is then proposed and verified with simulations. In Section 3, we discuss the fairness issues based on the throughput model. We will explain the origin of unfairness in the CSMA/CA network with simulation support, and discuss the trade-off between maximum throughput and fairness. The approximations of throughput with local topology information are presented and their performance is investigated in Section 4. We then discuss distributed fair scheduling based on throughput approximations in Section 5. Three different fairness metrics and their implementations are discussed. We conclude our paper in Section 6 and present the necessary proofs in the appendix.

Both authors are with Department of Electrical, Computer and System Engineering, Rensselaer Polytechnic Institute, NY 12180, US.
E-mail:wangx5@rpi.edu, koushik@ecse.rpi.edu.

II. Throughput Model of CSMA/CA Network

A. Overview of CSMA/CA Networks

The goal of the CSMA/CA protocol is to maximize throughput by reducing the collisions due to the contending nodes that share the same channel. Both physical channel sensing and virtual channel sensing are used to attain this goal.

Like CSMA, a node in the CSMA/CA network senses the channel before it transmits a frame. If the channel is idle, the node transmits its frame. Otherwise, it defers the transmission till the end of the ongoing transmission. It then randomly chooses a value, known as a contention window, denoted by cw that satisfies $CW_{min} \le cw \le CW_{max}$, and initializes its *backoff timer* with cw *backoff intervals*. Since cw is chosen randomly, the probability that two or more nodes choose the same backoff value is very low. The timer has the granularity of a *slot time* and is decremented by one every time the channel is sensed to be idle. The backoff timer is stopped in case the channel becomes busy and the decrementing process is resumed when the channel becomes idle again. The node is allowed to transmit its frame when the backoff timer reaches zero. Each successful transmission is positively acknowledged so that transport layer retransmission is avoided.

Due to a particular feature of wireless networks known as the "hidden node" problem, the physical channel sensing method alone cannot avoid collisions. Two nodes that are not within hearing distance of each other may still lead to collisions at a third node which receives the transmission from both sources. The problem is illustrated in Fig. 1 when both A and D transmit to B. To take care of this problem, CSMA/CA uses the Request to Send (RTS) and Clear to Send (CTS) control frames to reserve the channel. A node with a frame to transmit sends an RTS frame to the receiver and the receiver responds with a CTS frame if it is currently not busy. This RTS/CTS exchange, which also contains timing information about the length of the ensuing transmission, known as NAV (Network Allocation Vector), is detected by all the nodes within hearing range of either the sender or the receiver or both. These nodes defer their transmissions till the ongoing transmission is complete. This is the so called virtual channel sensing. Its use is shown in Fig. 2 when A in Fig. 1 transmits to B.

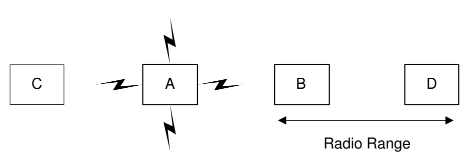

Fig. 1. Hidden node problem occurs when both A and D transmit to B.

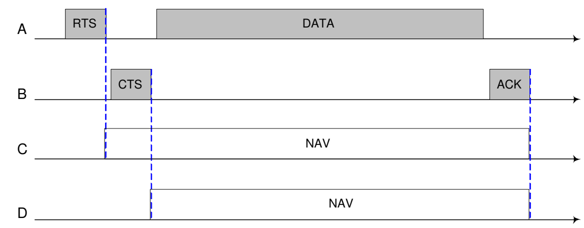

Fig. 2. The use of virtual channel sensing.

B. System Model and Assumptions

A general wireless ad-hoc network can be modelled as an undirected graph $G = (N, L)$, where N and L respectively denote the set of nodes and the set of (undirected) links. A link exists between two nodes if and only if they can hear each other (we assume a symmetric hearing matrix $\mathbf{H}$). A directed edge (i, j) represents an active communication pair, and E is the set of directed edges. Note that only a few pairs may be actively communicating although there are $2|L|$ possible communication pairs. Two-way error free connectivity is assumed in our model, i.e. a frame fails to be received if and only if it collides with other frames.

For any node i in the network, the set of i's *neighbors*, $\mathfrak{N}_i^* = \{j : (i, j) \in L\}$ represent the set of nodes that can hear i. Denote $\mathfrak{N}_i = \mathfrak{N}_i^* + \{i\}$. For a collection of nodes A, let $\mathfrak{N}(A) = \bigcup_{i \in A} \mathfrak{N}_i$.

For any link $(i, j) \in E$, $\mathfrak{B}(i, j)$ denotes the set of links blocked by link (i, j) and define $\mathfrak{B}^*(i, j) = \mathfrak{B}(i, j) - \{(i, j)\}$. For a collection of links A, define $\mathfrak{B}(A) = \bigcup_{e \in A} \mathfrak{B}(e)$. According to the CSMA/CA protocol, when link (i, j) is ongoing, neighbors of i receive RTS from i and neighbors of j receive CTS from j. These nodes will defer their transmissions and they won't issue an RTS or respond to a CTS. Therefore $\mathfrak{B}(i, j) = \{(s, t) : s \in \mathfrak{N}(i \bigcup j) \text{ or } t \in \mathfrak{N}(i \bigcup j), \forall (s, t) \in E\}$. Note that if $(s, t) \in \mathfrak{B}(i, j)$, we also have $(i, j) \in \mathfrak{B}(s, t)$.

We assume that each node has a single transceiver. Thus, a node can not transmit and receive simultaneously, and they cannot receive more than one frame at a time. If two (or more) transmissions are heard simultaneously by a node, at least one and possibly both transmitted frames are lost and must be retransmitted. For simplicity of study, we assume that a frame can be retransmitted as many times as is necessary.

We further assume that the scheduling point process for each used link $(i, j) \in E$ is Poisson with mean λ_{ij}, and is independent of all other such processes in the network. In addition, the frame lengths are assumed to be exponentially distributed, with mean transmission time $1/\mu_i$ for frames transmitted by node i.

The propagation delay is assumed to be zero in our model. The RTS and CTS are assumed to be very small, and their transmission time can be ignored. Acknowledgements are obtained instantaneously.

For simplicity, we assume that none of the nodes adopts the BEB (Binary Exponential Backoff) algorithm, i.e. the CW_{max} is fixed and won't be increased for retransmissions. This is reasonable, as each node is assumed to choose its maximum contention window size according to the network topology and hence dynamic adaption of the contention window size is no longer necessary.

To sum up, our assumptions are as follows:

1) The scheduling point process of a used link is Poisson, and is independent of all other such processes in the network.
2) Packet lengths are exponentially distributed and are generated independently at each transmission.

3) The propagation delay between neighboring nodes is zero.
4) Under CSMA/CA, a node will transmit a scheduled packet if both its physical channel sensing method and virtual channel sensing method detect an idle channel.
5) Links are error free. A packet will be successfully received if and only if there is no collision at the receiver side.
6) The lengths of RTS and CTS are very small and their transmission time can be ignored. Acknowledgements are obtained instantaneously.
7) The BEB algorithm is not deployed in the network and each link has a fixed CW_{max} value.

C. *Throughput Models*

In this section, we present the CSMA/CA throughput model based on Markovian analysis using an approach similar to that adopted in [2] for pure CSMA networks.

Under the assumptions discussed above, a scheduling point for link (i,j) results in a successful transmission if and only if all the links that belong to $\mathfrak{B}(i,j)$ are idle at the scheduling point of (i,j). The average time of such a successful transmission is simply $1/\mu_i$. Since the scheduling point is Poisson, each scheduling point is random if looked at in time. For any set of used links $A \subset E$, $P(A)$ denotes the probability that all links belonging to A are idle. According to the Poisson Arrivals See Time Averages Theorem, the throughput x_{ij} of link (i,j) can be expressed as

$$x_{ij} = \frac{\lambda_{ij}}{\mu_i} P(\mathfrak{B}(i,j)) \tag{1}$$

To compute $P(\mathfrak{B}(i,j))$, given the assumption that the hearing matrix is symmetric, it is sufficient to consider the process $\{X(t), t \geq 0\}$, where $X(t)$ is the set of links that are busy transmitting at time t. Under the assumptions of exponential message lengths and Poisson scheduling point process, $X(t)$ is a continuous time Markovian chain, and the state for the Markovian chain $X(t)$ at time instate t is the *independent set*, the set of links that can be active at the same time. We denote the state space of $X(t)$ as $\mathfrak{S}$, which can be determined from the hearing matrix $\mathbf{H}$ of the network.

Let $\mathcal{P}(Q)$ be the steady-state probability of the state Q. As $\mathfrak{B}(Q)$ denotes all the links that are blocked when any link in Q is active, $Q + \{(i,j)\}$ and $Q - \{(i,j)\}$ denote the state formed by adding link (i,j) to the state Q or removing link (i,j) from state Q respectively. Assuming that the network operation is stable, the steady-state probability satisfies the following balance equations:

$$\begin{aligned}&\mathcal{P}(Q)\left[\sum_{(i,j)\in Q} \mu_i + \sum_{(i,j)\notin \mathfrak{B}(Q)} \lambda_{ij}\right] = \\ &\sum_{(i,j)\notin \mathfrak{B}(Q)} \mu_i \mathcal{P}(Q+\{(i,j)\}) + \sum_{(i,j)\in Q} \lambda_{ij}\mathcal{P}(Q-\{(i,j)\})\end{aligned} \tag{2}$$

Define the scheduling rate of link (i,j) as $\rho_{ij} = \lambda_{ij}/\mu_i$. It is easy to verify that (2) is consistent with the detailed balance equations

$$\mathcal{P}(Q+\{(i,j)\}) = \rho_{ij}\mathcal{P}(Q) \tag{3}$$

Therefore

$$\mathcal{P}(Q) = \left(\prod_{(i,j)\in Q} \rho_{ij}\right)\mathcal{P}(\phi) \tag{4}$$

constitutes the solution to (2). The constant $\mathcal{P}(\phi)$ is obtained by normalizing the distribution, i.e.

$$\mathcal{P}(\phi) = \left[\sum_{Q\in\mathfrak{S}} \prod_{(i,j)\in Q} \rho_{ij}\right]^{-1} \tag{5}$$

Given the product form formulation (4), the probability that all used links in the set A are idle, $P(A)$, can be found by summing $\mathcal{P}(Q)$ over all independent sets Q that do not contain any link in A. Thus

$$P(A) = \sum_{Q\subset A^c} \mathcal{P}(Q) = \frac{\sum_{Q\subset A^c}\left(\prod_{(i,j)\in Q} \rho_{ij}\right)}{\sum_{Q\in\mathfrak{S}}\left(\prod_{(i,j)\in Q} \rho_{ij}\right)} \tag{6}$$

where $Q \subset A^c$ refers to all independents sets contained in the complement of A. We adopt the shorthand notation

$$\Psi(A) = \sum_{Q\subset A}\left(\prod_{(i,j)\in Q} \rho_{ij}\right) \tag{7}$$

and define $\Psi(\phi) = 1$. Therefore

$$P(A) = \frac{\Psi(A^c)}{\Psi(\mathfrak{S})} \tag{8}$$

Using the shorthand notation, (1) can be written as

$$x_{ij} = \rho_{ij}\frac{\Psi\left(\mathfrak{B}(i,j)^c\right)}{\Psi\left(\mathfrak{S}\right)} \tag{9}$$

Denote the length of the data frame from node i as L_i. As the length of RTS/CTS and ACK is assumed to be very small, μ_i can be estimated by

$$\mu_i = C/L_i \tag{10}$$

where C is the link capacity. Denote the slot time for the contention window as T. The contention window cw is a random variable uniformly distributed between CW_{min} and CW_{max} according to the CSMA/CA protocol, its expected value is $E(cw) = 0.5(CW_{min} + CW_{max})$, and λ_{ij} can be estimated by

$$\lambda_{ij} = \frac{1}{E(cw)T} = \frac{2}{(CW_{min} + CW_{max})T} \tag{11}$$

Therefore ρ_{ij} is

$$\rho_{ij} = \frac{\lambda_{ij}}{\mu_i} = \frac{2L_i}{(CW_{min} + CW_{max})CT} \tag{12}$$

Substituting (12) into (9), we can compute the throughput for each link (i,j) in the CSMA/CA network.

In the cases when the data frame size is not large enough and the transmission time of RTS/CTS and ACK cannot be ignored, we can introduce a correction factor to take into account the overhead of RTS/CTS and ACK. The corrected throughput for link (i, j) is

$$x_{ij} = \rho_{ij} \frac{\Psi\left(\mathfrak{B}(i,j)^c\right)}{\Psi\left(\mathfrak{S}\right)} \cdot \frac{L_i}{L_i + H} \tag{13}$$

where H is the overhead size of the RTS/CTS and ACK control frames. In computing ρ_{ij}, the overhead of RTS/CTS and ACK also should be taken into consideration, and therefore

$$\rho_{ij} = \frac{2(L_i + H)}{(CW_{min} + CW_{max})CT} \tag{14}$$

D. Simulation Investigation

In this section, we present simulation results to investigate the performance of the network throughput model. Although only the simulation results for the simple three-link scenario are presented in this paper, simulations carried out on various network topologies/scenarios give similar results. Refer to [11] for more simulation investigations.

1) Simulation Setup: Instead of using ns-2, we realize a simple event driven simulator to investigate the performance of the CSMA/CA network throughput model. This is due to the fact that ns-2 involves all network layers, and furthermore, the CSMA/CA model in ns-2 does not reset NAV [1] and might lead to false channel reservation in certain situations.

Our simulator is essentially a simplified version of the wireless LAN implementation in ns-2, but it contains the MAC layer implementation only and provides the NAV reset function. The frame sizes of RTS, CTS and ACK are set according to the 802.11 specification [1], but none of the inter-frame spaces is considered in our simulator, i.e. all inter-frame spaces are set to zero.

As CSMA/CA usually deploys the BEB algorithm to adapt the maximum contention window to the contending nodes, the maximum contention window size is fixed in our simulator. However, different links may choose different CW_{max} values.

2) Three-Link Scenario: In this section, we present the simulation results of a simple yet illustrative three-link scenario.

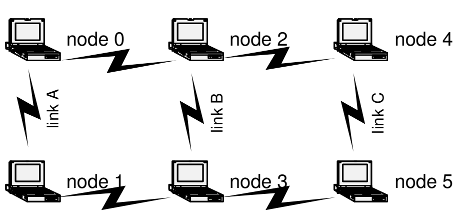

Fig. 3. Three-Link Network Topology.

The network topology is shown in Fig. 3. There are 6 nodes and 7 links, but only link $(0, 1)$ (denoted as link A), link $(2, 3)$ (denoted as link B) and link $(4, 5)$ (denoted as link C) are active transmission pairs. Due to symmetry, link A and link C always have the same performance. So each time we consider the network performance, we only consider link A and link B.

Note that when a transmission is ongoing on link B, neither link A nor link C can be scheduled. However, link A and link C can be scheduled at the same time. The state transition diagram is shown in Fig. 4.

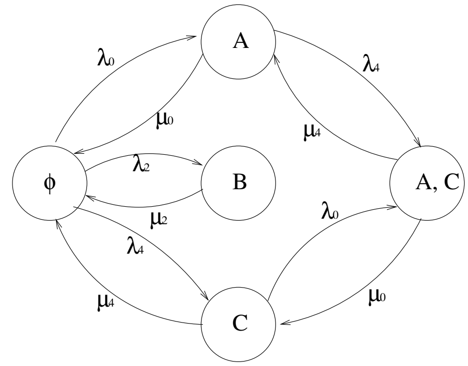

Fig. 4. State Transition Diagram.

In our simulations, all the links have the same data frame size and same CW_{max} and CW_{min}, therefore $\rho_A = \rho_B = \rho_C = \rho$. According to (9) the throughput of link A is

$$\begin{aligned} x_A &= \rho_A \left(\mathcal{P}(\phi) + \mathcal{P}(C)\right) = \rho_A \frac{\Psi(C)}{\Psi(A,B,C)} \\ &= \frac{\rho_A + \rho_A\rho_C}{1 + \rho_A + \rho_B + \rho_C + \rho_A\rho_C} = \frac{\rho + \rho^2}{1 + 3\rho + \rho^2} \end{aligned} \tag{15}$$

Similarly, the throughput of link B and link C are

$$x_B = \frac{\rho_B}{1 + \rho_A + \rho_B + \rho_C + \rho_A\rho_C} = \frac{\rho}{1 + 3\rho + \rho^2} \tag{16}$$

$$x_C = \frac{\rho_C + \rho_A\rho_C}{1 + \rho_A + \rho_B + \rho_C + \rho_A\rho_C} = \frac{\rho + \rho^2}{1 + 3\rho + \rho^2} \tag{17}$$

where ρ is estimated using (12), $CW_{min} = 0$, $T = 20\mu sec$, and $C = 1$Mbps.

Fig. 5 plots the estimated throughput and the simulated throughput of link A and link B when the data frame size is fixed to 500 bytes and different CW_{max} values and hence different ρ values are chosen. The propagation delay between neighboring nodes is $1\mu s$, and the length of RTS/CTS and ACK is 48 bytes in total.

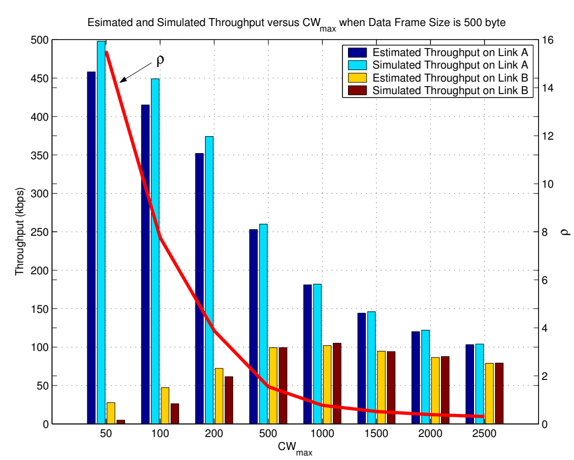

Fig. 5. Estimated Throughput and Simulated Throughput versus CW_{max} When Data Frame Length is 500 bytes in the Three-Link Scenario.

From Fig. 5, we can see that when ρ is small (close to 1 or smaller than 1) the computed throughput values based on our model match the simulation results perfectly. It is interesting to note that the difference between the estimated throughput and the simulated throughput is quite large when CW_{max} is small. This is due to the fact that our throughput formulation in (1) is based on the assumption of Poisson arrival of the scheduling point of the used links. When CW_{max} is small, this assumption does not hold and hence our throughput formulation is not accurate. In the situations when CW_{max} is comparatively large, the frames necessary for retransmissions are rescheduled after a sufficiently long randomized time and the Poisson arrival assumption roughly holds. This can be verified from Fig. 5, as the estimated throughput based on our formulation is very close to the simulated throughput when CW_{max} is large.

3) Throughput Correction When Data Frame is Small: According to 802.11 specifications, RTS is 20 bytes, while both CTS and ACK are 14 bytes. Therefore the overhead from RTS/CTS and ACK is 48 bytes in total. When the data frame size is small, the length of RTS/CTS and ACK can no longer be ignored. In this section, we investigate the performance of the corrected network throughput formula in (13) and (14).

We still consider the three-link scenario in Fig. 3. In order to observe the difference between the corrected throughput and non-corrected throughput, we choose a relatively small data frame size. In the simulations, the data frame size is fixed to 100 bytes while different CW_{max} values are chosen. The simulated throughput of link A and link B, the computed throughput with and without corrections under different CW_{max} values are presented in Table I and Table II respectively.

TABLE I
THE SIMULATED THROUGHPUT, THE COMPUTED THROUGHPUT WITH AND WITHOUT CORRECTION FOR LINK A

CW_{max}	Simulated throughput (kbps)	Throughput with correction (kbps)	Throughput with no correction (kbps)
50	3.597×10^5	3.930×10^5	4.976×10^5
100	2.795×10^5	2.935×10^5	3.564×10^5
200	2.001×10^5	2.037×10^5	2.373×10^5
500	1.115×10^5	1.120×10^5	1.233×10^5

TABLE II
THE SIMULATED THROUGHPUT, THE COMPUTED THROUGHPUT WITH AND WITHOUT CORRECTION FOR LINK B

CW_{max}	Simulated throughput (kbps)	Throughput with correction (kbps)	Throughput with no correction (kbps)
50	1.235×10^5	1.167×10^5	1.914×10^5
100	1.346×10^5	1.344×10^5	1.980×10^5
200	1.258×10^5	1.279×10^5	1.695×10^5
500	9.150×10^4	9.058×10^4	1.063×10^5

From Table I and Table II, it is obvious that the correction greatly improves the throughput modelling accuracy. When CW_{max} is small the improvement is significant.

III. FAIRNESS ISSUES IN CSMA/CA NETWORKS

In this section, we consider the fairness problem in CSMA/CA networks. We will analyze the origin of the unfairness and present simulation results to support our analysis.

A. Unfairness in CSMA/CA Networks

Heuristically, the hidden node problem in wireless networks causes unfairness. This can be illustrated by considering the three-link scenario shown in Fig. 3.

In this scenario, a possible situation is as follows. After node 0 grasps the channel and link A is scheduled, there is some data to be transmitted on link B. As node 2 senses a busy channel, it starts the backoff timer for link B and begins to wait. However, node 4 will sense an idle channel and link C can be scheduled immediately if there is data to be transmitted on link C. At the time when transmission on link A stops, the transmission on link C may be still active and hence node 2 still senses a busy channel and link B still has to wait. Before the transmission on link C ends, transmission on link A may seize the channel again, and before transmission on link A ends, transmission on link C is active again, and so on. In this way, link A and link C access the channel in turn while link B may never be able to find a chance to be scheduled for transmission.

The unfairness in this scenario can be easily shown by considering the throughput formula in (15), (16) and (17). When the three links have the same scheduling rate ρ, we can easily see that $\frac{x_A}{x_B} = \frac{x_C}{x_B} = 1 + \rho$. If ρ is large, the throughput of link A and link C will be much larger than the throughput of link B and hence lead to serious fairness problems. In fact, when ρ approaches infinity, x_A and x_B approach to 1 while x_B approaches to 0, i.e. transmissions on link A and link C always can be carried out while B finds no chance to be scheduled.

B. Binary Exponential Backoff Algorithm Aggravate Fairness Problem in CSMA/CA Networks

In CSMA/CA protocol, the BEB (Binary Exponential Backoff) algorithm is often adopted so that a node can dynamically adjust its maximum contention window size to the number of contending nodes. As all the nodes in the network employ this algorithm, it seems that all the contending nodes will have the same scheduling rate if they have the same data frame size. However, this is not the case.

Consider the simple scenario in Fig. 3. As link B contends with both link A and link C, node 2 has a higher probability to sense a busy channel than node 0 or node 4. Therefore, if all the nodes in the network employ the BEB Algorithm, node 2 definitely will execute the BEB algorithm much more often than node 0 and node 4. As a result, the maximum contention window size of link B will be much larger than that of link A and link C, and the scheduling rate of link B will be much lower than that of link A or link C. Intuitively, this means that link B has to wait for a longer time to sense the channel again while link A and C can sense the channel much more quickly. Thus the employment of BEB algorithm in the MAC layer indeed aggravates the unfairness in the CSMA/CA network.

C. Simulation Investigation

In this section we investigate the fairness issues in the three-link scenario with simulations.

We use the ratio of throughput of link A to that of link B as the metric to measure the fairness between these two links. Fairness is good when the ratio is close to 1, and is extremely bad when the ratio approaches infinity or approaches zero.

The data frame size in our simulation is fixed to 500 bytes, and different CW_{max} and hence different ρ values are chosen. The ratio of throughput on link A to that on link B is investigated. The simulation results are plotted in Fig. 6. The corresponding ρ values are also plotted.

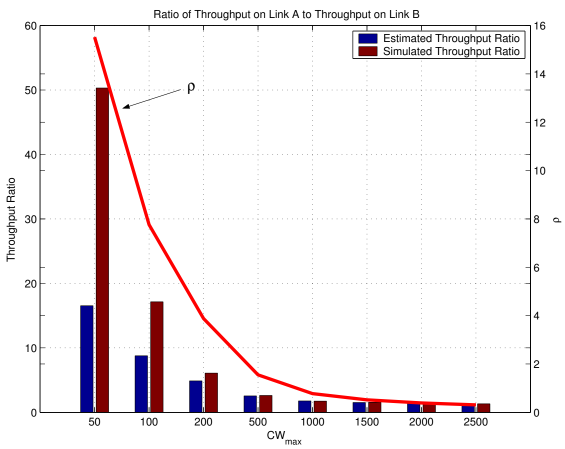

Fig. 6. Ratio of Throughput on Link A to Throughput on Link B without BEB.

Based on the throughput model of CSMA/CA network, we have shown in the last section that the throughput ratio is greatly larger than 1 and hence the fairness performance is extremely poor when ρ is large, while the throughput ratio is very close to 1 and a satisfactory fairness performance is achieved when ρ is small. The simulated throughput ratios state the same fact, except that the fairness performance is even worse when ρ is large. The mismatch comes from the fact that our throughput model is not so accurate when the Poisson arrival assumption does not hold.

We then investigate how the BEB algorithm impacts the fairness in the three-link scenario. To gain a better view on what is going on, we investigate the total amount of data transmitted on the three links versus time.

The simulation runs 100 seconds, and the total amount of data transmitted on link A, link B and link C are plotted in Fig. 7. A serious fairness problem is obvious from the plot: the total amount of data transmitted on link B remains a flat line from time to time, which means that no transmission occurs on link B during those time periods. At the 76th second, link B is idle for the following 15 seconds. At the end of the simulation, the data transmitted through link A and link C are 6,800 KBytes each, while the data transmitted through link B are only 18 KBytes. The amount of data received through link A or link C is therefore about 380 times that through link B.

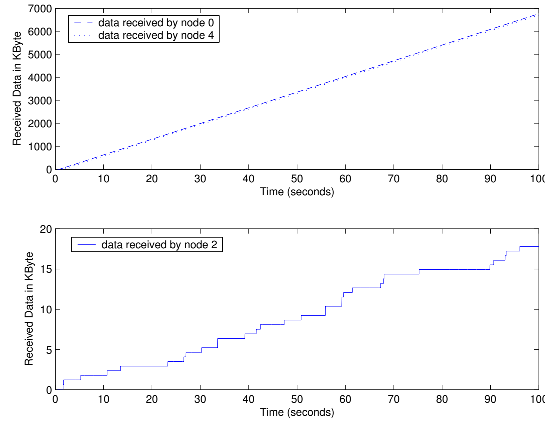

Fig. 7. The amount of data received on link A, link B and link C respectively with BEB.

D. Trade-Off Between Throughput and Fairness

There is a trade off between the total throughput of the CSMA/CA network and the fairness provided among all the active links in the network. To see this, we still assume that all the links have the same scheduling rate.

We have already shown that the throughput of link (i, j) is

$$x_{ij} = \rho \frac{\Psi(\mathfrak{B}(i,j)^c)}{\Psi(\mathfrak{S})}$$

It is worth noting that $\phi \subset \mathfrak{B}(i,j)^c$, and hence $\Psi(\mathfrak{B}(i,j)^c)$ approaches to 1 when ρ approaches zero. Similarly, as $\phi \subset \mathfrak{S}$, $\Psi(\mathfrak{S})$ also approaches to 1 when ρ approaches zero. Therefore when ρ is very small

$$x_{ij} = \rho \frac{\Psi(\mathfrak{B}(i,j)^c)}{\Psi(\mathfrak{S})} \approx \rho \tag{18}$$

Equation (18) states the fact that the throughput of a link approaches to the scheduling rate when the scheduling rate approaches zero. This makes sense intuitively, as a transmission will have a lower probability of a collision if other links have very low scheduling rates. In this case, ideal fairness is attained if all the links have the same small scheduling rate. However, this case is not ideal as the throughput, which is equal to the scheduling rate, is also very small and hence the utilization of the network is very low.

We illustrate this case with the three-link scenario. In this scenario, the throughput for the three links are

$$x_A = x_C = \frac{\rho + \rho^2}{1 + 3\rho + \rho^2}, \quad x_B = \frac{\rho}{1 + 3\rho + \rho^2}$$

When ρ is very small, $x_A \approx x_B \approx x_C \approx \rho$ and hence the fairness among the three links is guaranteed while the total throughput is very low.

We then consider the case when ρ approaches infinity. Denote the maximum independent set that contains link (i, j) as $S_{(i,j)}$ and its degree as M. Define $S^*_{(i,j)} = S_{(i,j)} - \{(i, j)\}$. It follows that $\Psi(\mathfrak{B}(i,j)^c) = \Theta(\rho^{M-1})$, since $S^*_{(i,j)} \subset \mathfrak{B}(i,j)^c$ and $S^*_{(i,j)} \in \mathfrak{S}$ if $S_{(i,j)}$ is an independent set that contains link (i, j).

Denote the degree of the maximum independent set in the state space $\mathfrak{S}$ as N, it follows that $\Psi(\mathfrak{S}) = \Theta(\rho^N)$. Therefore

$$x_{ij} = \rho \frac{\Psi(\mathfrak{B}(i,j)^c)}{\Psi(\mathfrak{S})} \approx \Theta(\rho^{M-N}) \tag{19}$$

.

If link (i, j) belongs to any of the maximum independent set of $\mathfrak{S}$, then $N = M$, and x_{ij} approaches a positive constant when ρ approaches infinity. If link (i, j) does not belong to any of the maximum independent set of $\mathfrak{S}$, it follows that $M < N$ and hence $\lim_{\rho\to\infty} x_{ij} = 0$.

Therefore when ρ approaches infinity, only the links that belong to the maximum independent set have a throughput greater than zero and the throughput of all other links is zero. In this case, maximum throughput is achieved while fairness performance is extremely bad.

We use the three-link scenario to illustrate this case. As

$$x_A = x_C = \frac{\rho + \rho^2}{1 + 3\rho + \rho^2}, \quad x_B = \frac{\rho}{1 + 3\rho + \rho^2}$$

When ρ approaches infinity, we have

$$x_A = x_C \approx 1 \quad x_B \approx 0$$

which matches our prediction.

IV. Throughput Approximations with Local Information

To provide a satisfactory fairness performance in the CSMA/CA based network, a naive approach is to choose the scheduling rate wisely for each used link so that the desired fairness criteria, e.g. the max-min fairness or proportional fairness, can be fulfilled. However, this task remains formidable for a network with a general topology, as finding all the independent sets is an NP hard problem. Moreover, finding such sets needs the topology information of the entire network and is infeasible for the distributed MAC layer implementation.

A. Throughput Approximation with Local Information

In this section, we discuss the possibility of approximating the throughput with local topology information.

Before discussing the throughput approximation, we present an important property involving two sets that have no element in common.

Theorem 1: If $A \cap B = \phi$, $\Psi(A \cup B) \le \Psi(A)\Psi(B)$.

The proof of Theorem 1 can be found in the appendix.

Using Theorem 1, we have the following corollary.

Corollary 1: If we denote

$$u_{ij} = \frac{\rho_{ij}}{\Psi(\mathfrak{B}(i,j))} \tag{20}$$

the throughput of link (i, j), x_{ij}, satisfies $x_{ij} \ge u_{ij}$.

The proof of *Corollary 1* is straightforward, and is provided in the appendix.

From *Corollary 1* we can see that u_{ij} is indeed a lower bound for the throughput on link (i, j). Note that u_{ij} involves $\mathfrak{B}(i, j)$ only and can be computed with local topology information within the two-hop neighborhood of both node i and j.

Noting that $\prod_{(s,t)\in\mathfrak{B}(i,j)}(1+\rho_{st})$ corresponds to the power set of $\mathfrak{B}(i, j)$ and hence $\prod_{(s,t)\in\mathfrak{B}(i,j)}(1+\rho_{st}) \ge \Psi(\mathfrak{B}(i, j))$, we can derive another approximation of x_{ij}.

Corollary 2: If we denote

$$v_{ij} = \frac{\rho_{ij}}{\prod\limits_{(s,t)\in\mathfrak{B}(i,j)} (1 + \rho_{st})} \tag{21}$$

the throughput of link (i, j), x_{ij}, satisfies $x_{ij} \ge v_{ij}$.

Regarding to the performance of u_{ij} and v_{ij}, we have the following corollary.

Corollary 3: When ρ_{ij} is small, both throughput approximations, u_{ij} and v_{ij}, are very close to the throughput x_{ij}. Specifically,

$$\lim_{\rho_{st}\to 0, \forall(s,t)\in E} \frac{u_{ij}}{x_{ij}} = 1 \qquad \lim_{\rho_{st}\to 0, \forall(s,t)\in E} \frac{v_{ij}}{x_{ij}} = 1$$

The proof of *Corollary 3* is in the appendix.

B. Simulation Investigation of the Throughput Approximation Performance

In this section, we investigate the performance of the two approximations, namely u_{ij} and v_{ij}.

We consider the three-link network scenario and assume that all the links have the same scheduling rate. As $\mathfrak{B}(A) = \{A, B\}$ in this scenario, we have

$$u_A = \frac{\rho_A}{\Psi(A,B)} = \frac{\rho_A}{1 + \rho_A + \rho_B} = \frac{\rho}{1 + 2\rho}$$

$$v_A = \frac{\rho_A}{\prod\limits_{i\in\{A,B\}} (1 + \rho_i)} = \frac{\rho_A}{(1 + \rho_A)(1 + \rho_B)} = \frac{\rho}{(1 + \rho)^2}$$

Similarly, $\mathfrak{B}(B) = \{A, B, C\}$, and $\mathfrak{B}(C) = \{B, C\}$. Therefore

$$u_B = \frac{\rho}{1 + 3\rho + \rho^2} \qquad v_B = \frac{\rho}{(1 + \rho)^3}$$
$$u_C = \frac{\rho}{1 + 2\rho} \qquad v_C = \frac{\rho}{(1 + \rho)^2}$$

We investigate the performance of the approximation of u_{ij} and v_{ij} when ρ is small and when ρ is large respectively. The throughput on link A and its approximations versus ρ are plotted in Fig. 8, and throughput on link B and its approximations are plotted in Fig. 9.

From the simulation results it can be seen that both approximations, u_{ij} and v_{ij}, can approximate the throughput well when ρ is small. When ρ is large, both u_{ij} and v_{ij} are not good approximations for throughput x_{ij}. Despite the fact that the performance of u_{ij} is much better than that of v_{ij}, which is especially obvious for link A as v_A approaches zero, the difference between the approximation u_{ij} and the throughput x_{ij} is still significant.

V. Distributed Fairness Scheduling in CSMA/CA Based Ad Hoc Networks

As mentioned in the last section, throughput computation involves knowledge of the entire network topology. Therefore fairness guarantee of throughput cannot be attained in the distributed manner.

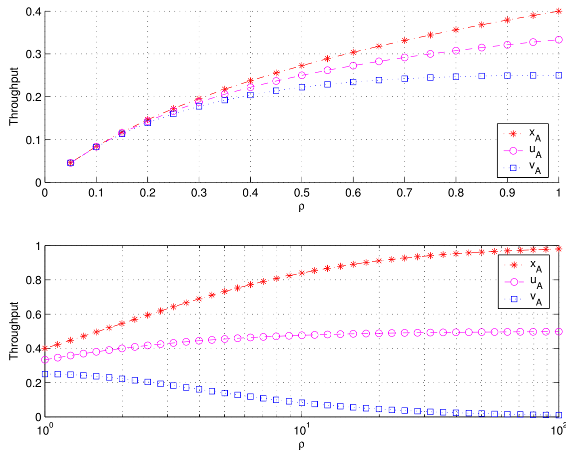

Fig. 8. Throughput of link A and its approximation versus ρ.

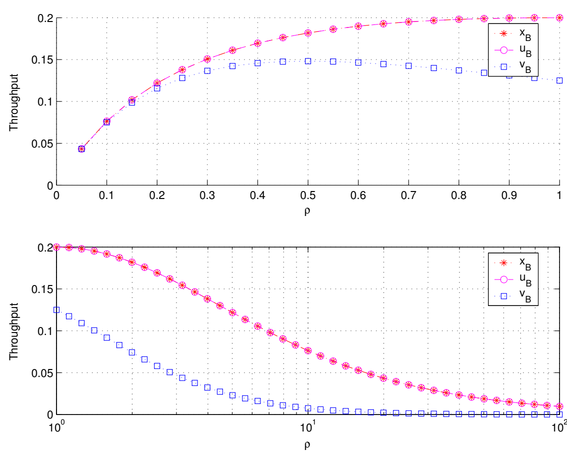

Fig. 9. Throughput of link B and its approximation versus ρ (Note that $x_B = u_B$ and hence the two curves overlap).

An alternative approach to this problem is that, instead of providing fairness for throughput, we provide fairness based on the throughput approximation, i.e. u_{ij} or v_{ij}. As throughput approximation involves only local topology information, it is feasible that a distributed fair scheduling method can be developed based on throughput approximation.

Note that when ρ is small, both u_{ij} and v_{ij} are good approximations of x_{ij}. Although u_{ij} is better than v_{ij}, v_{ij} has a much simpler form and hence is more likely to lead to an easy implementation of distributed fair scheduling methods. Therefore we use v_{ij} instead of u_{ij} in providing fairness in a distributed manner.

A. Proportional Fairness Scheduling

In this section, we discuss how to choose ρ_{ij} so that proportional fairness amongst the throughput approximation v_{ij} can be achieved.

Let $\vec{\rho}$ denote the vector of ρ_{ij} for all used links. The problem can be formulated as

$$\vec{\rho}^{*} = \arg\max_{\rho_{ij}>0} \sum_{(i,j)\in E} \log\left(v_{ij}(\vec{\rho})\right) \tag{22}$$

where $v_{ij}(\vec{\rho})$ is the approximation for the throughput x_{ij}.

The following result shows how ρ_{ij} for a used link (i,j) can be computed based on local topology information (The proof is in the appendix).

Theorem 2: $\vec{\rho}^{*}$, the optimal scheduling rate on any used link $(i,j) \in E$, is given by

$$\rho^{*}_{ij} = \begin{cases} \dfrac{1}{|\mathfrak{B}^{*}(i,j)|} & \text{when } \mathfrak{B}^{*}(i,j) \neq \phi \\ +\infty & \text{when } \mathfrak{B}^{*}(i,j) = \phi \end{cases} \tag{23}$$

where $|\mathfrak{B}^{*}(i,j)|$ denotes degree of the set $\mathfrak{B}^{*}(i,j)$.

Note that $|\mathfrak{B}^{*}(i,j)|$ denotes the total number of links blocked by link (i,j) except itself, thus the scheduling rate of link (i,j) is inversely proportional to the total number of links it's contending with.

This information can be obtained by local information exchange. When the network is formed, or when the network topology changes, each node can broadcast information on the links out of itself and the link into itself to all nodes within two hops. The scheduling rate can then be set accordingly.

B. Two-Hop Fairness Scheduling

In this section, we discuss another fairness guarantee, called "Two-Hop Fairness", in a CSMA/CA based ad-hoc network. This is so called, as each used link (i,j) in the network will be guaranteed a minimum throughput in a form depending on a factor decided by the topology information within two hops of link (i,j).

The throughput approximation v_{ij} can be rewritten as

$$v_{ij} = \frac{\rho_{ij}}{1+\rho_{ij}} \cdot \frac{1}{\prod\limits_{(s,t)\in\mathfrak{B}^{*}(i,j)} (1+\rho_{st})}$$

It can be seen that v_{ij} is a monotonously increasing function for ρ_{ij} and monotonously decreasing function for ρ_{st} where link (s,t) belongs to $\mathfrak{B}^{*}(i,j)$. Intuitively, the term $\frac{\rho_{ij}}{1+\rho_{ij}}$ can be viewed as the scheduling rate of link (i,j), while the term $1/\prod_{(s,t)\in\mathfrak{B}^{*}(i,j)}(1+\rho_{st})$ can be viewed as the probability that such a scheduling results in a successful transmission. If $|\mathfrak{B}^{*}(i,j)|$ is large, the probability that a scheduling of link (i,j) is successful will be low and hence the throughput will be low. In order to have a high throughput for link (i,j), it is important that the links that block (i,j) have low scheduling rates. Put it in another way, although larger ρ_{ij} will lead to higher throughput of (i,j), it will also lead to lower throughput for the links that belong to $\mathfrak{B}(i,j)$, which is not desirable from the view of the system.

A reasonable way of choosing ρ_{ij} is $\rho_{ij} = 1/\Delta_{ij}$, where $\Delta_{ij} = \max_{(s,t)\in\mathfrak{B}^{*}(i,j)}\{|\mathfrak{B}^{*}(s,t)|\}$.

Noting that $\Delta_{st} \geq |\mathfrak{B}^{*}(i,j)|$ for any link (s,t) that belongs to $\mathfrak{B}^{*}(i,j)$, we have

$$\begin{aligned} v_{ij} &= \frac{\rho_{ij}}{1+\rho_{ij}} \cdot \frac{1}{\prod\limits_{(s,t)\in\mathfrak{B}^{*}(i,j)} (1+\rho_{st})} \\ &= \frac{1/\Delta_{ij}}{1+1/\Delta_{ij}} \cdot \frac{1}{\prod\limits_{(s,t)\in\mathfrak{B}^{*}(i,j)} (1+1/\Delta_{st})} \\ &\geq \frac{1}{\Delta_{ij}+1} \cdot \frac{1}{(1+\frac{1}{|\mathfrak{B}^{*}(i,j)|})^{|\mathfrak{B}^{*}(i,j)|}} \geq \frac{1}{e\,(\Delta_{ij}+1)} \end{aligned} \tag{24}$$

Therefore we can guarantee some minimum throughput guarantee for link (i,j) by choosing ρ_{ij} in this manner.

C. Max-Min Fairness Scheduling

For link (i,j), denote $p_{ij} = \rho_{ij}/(1+\rho_{ij})$, and denote $\mathbf{p}$ as the vector of p_{ij} for all used links. Then the throughput approximation v_{ij} can be rewritten as

$$v_{ij} = p_{ij} \prod_{(s,t)\in\mathfrak{B}^*(i,j)} (1-p_{st}) \tag{25}$$

Note that (25) is very similar in the form to the throughput of an ALOHA link [13]. Therefore we can provide max-min fairness for v_{ij} in the same way as we did in ALOHA networks [10].

1) Problem Formulation: We address the problem of how each active link should decide its scheduling rate ρ_{ij} so that the max-min link rate approximation in the CSMA/CA based ad-hoc network is maximized.

The max-min rate allocation problem for the rate approximation is formulated as

$$\begin{array}{ll} \max & v \\ \text{subject to} & v \le p_{ij} \prod\limits_{(s,t)\in\mathfrak{B}^*(i,j)} (1-p_{st}) \qquad \forall (i,j)\in E \end{array} \tag{26}$$

where v is the max-min fair rate approximation, and $p_{ij} = \rho_{ij}/(1+\rho_{ij})$.

We define a directed graph, called the link graph $G_L = (V_L, E_L)$, for the rate approximation v_{ij} in the CSMA/CA network, where each vertex stands for a link in the CSMA/CA network. There is an edge from link (i,j) to link (s,t) in the link graph if and only if $(s,t)\in\mathfrak{B}(i,j)$. As mentioned before, when $(s,t)\in\mathfrak{B}(i,j)$, we also have $(i,j)\in\mathfrak{B}(s,t)$. Therefore, the directed graph is bidirectional, i.e. if there is an edge from (i,j) to (s,t) in the link graph, there is also an edge from (s,t) to (i,j). Without loss of generality, we assume that the link graph for the CSMA/CA network is connected, as otherwise we can form a subproblem for each connected subgraph in the link graph and solve them separately. Also note that when the link graph is assumed to be connected, it is strongly connected since it's bidirectional.

The max-min fair rate allocation problem in (26) appears to be a non-convex problem. However, the following theorem(proof in Appendix) states that the max-min fair rates for v_{ij} in a CSMA/CA based ad-hoc network can be obtained by solving a convex optimization problem.

Theorem 3: The max-min fair rate allocation for v_{ij} in the CSMA/CA network can be formulated as the following convex programming problem

$$\begin{array}{ll} \min & \frac{1}{2}\sum\limits_{(i,j)\in E} f_{ij}^2 \\ \text{subj. to} & f_{ij} - \log(p_{ij}) - \sum\limits_{(s,t)\in\mathfrak{B}^*(i,j)} \log(1-p_{st}) \le 0 \\ & \qquad\qquad \forall (i,j)\in E \\ & f_{ij} = f_{st} \qquad \forall (i,j)\in E, \forall (s,t)\in\mathfrak{B}(i,j) \end{array} \tag{27}$$

where $f_{ij} = \log v_{ij}$.

2) Dual Method: The problem in (27) can be solved in a distributed manner using the dual method. We replace the constraints $f_{ij} = f_{st}$ in (27) by two inequality constraints, $f_{ij} \le f_{st}$ and $f_{st} \le f_{ij}$. The Lagrange function is then given by

$$\begin{aligned} L &= \frac{1}{2}\sum_{(i,j)\in E} f_{ij}^2 + \sum_{(i,j)\in E}\sum_{(s,t)\in\mathfrak{B}^*(i,j)} \lambda_{ij,st}(f_{ij}-f_{st}) \\ &+ \sum_{(i,j)\in E} \mu_{ij}\left(f_{ij} - \log p_{ij} - \sum_{(s,t)\in\mathfrak{B}^*(i,j)} \log(1-p_{st})\right) \end{aligned} \tag{28}$$

where μ_{ij} and $\lambda_{ij,st}$ are Lagrange multipliers.

Denote $\mathbf{f} = (f_{ij}, (i,j)\in E)$, $\boldsymbol{\mu} = (\mu_{ij}, (i,j)\in E)$, and $\boldsymbol{\lambda} = (\lambda_{ij,st}, (i,j)\in E, (s,t)\in\mathfrak{B}^*(i,j))$. The dual function is

$$D(\boldsymbol{\mu},\boldsymbol{\lambda}) = \min_{\mathbf{p},\mathbf{f}} L(\mathbf{p},\mathbf{f},\boldsymbol{\mu},\boldsymbol{\lambda}) \tag{29}$$

and the dual problem is

$$\text{D:} \quad \max_{\boldsymbol{\mu},\boldsymbol{\lambda}} D(\boldsymbol{\mu},\boldsymbol{\lambda}) \tag{30}$$

Convex duality implies that at the optimum $\boldsymbol{\lambda}^*$ and $\boldsymbol{\mu}^*$, the corresponding $\mathbf{p}$ and $\mathbf{f}$ are exactly the optimal solutions to the primal problem.

As the Lagrange function is concave for $\mathbf{p}$ and $\mathbf{f}$, the dual function can be solved by considering its partial derivatives, which gives

$$p_{ij}(\boldsymbol{\mu},\boldsymbol{\lambda}) = \frac{\mu_{ij}}{\sum\limits_{(s,t)\in\mathfrak{B}(i,j)} \mu_{st}} \tag{31}$$

and

$$f_{ij}(\boldsymbol{\mu},\boldsymbol{\lambda}) = \begin{cases} -\left(\mu_{ij} + \sum_{(s,t)\in\mathfrak{B}^*(i,j)} (\lambda_{ij,st} - \lambda_{st,ij})\right) \\ \text{if } \mu_{ij} + \sum_{(s,t)\in\mathfrak{B}^*(i,j)} (\lambda_{ij,st} - \lambda_{st,ij}) \ge 0 \\ 0 \\ \text{if } \mu_{ij} + \sum_{(s,t)\in\mathfrak{B}^*(i,j)} (\lambda_{ij,st} - \lambda_{st,ij}) < 0 \end{cases} \tag{32}$$

The dual problem can then be solved using gradient projection method, where the Lagrange multipliers are adjusted in the direction of the gradient $\nabla D(\boldsymbol{\mu},\boldsymbol{\lambda})$:

$$\lambda_{ij,st}(n+1) = \left[\lambda_{ij,st}(n) + \gamma_n \frac{\partial D}{\partial \lambda_{ij,st}}\right]^+ \tag{33}$$

and

$$\mu_{ij}(n+1) = \left[\mu_{ij}(n) + \gamma_n \frac{\partial D}{\partial \mu_{ij}}\right]^+ \tag{34}$$

Here $\gamma_n > 0$ is the step size at the nth iteration, and $[z]^+ = \max\{z,0\}$. The gradients are $\frac{\partial D}{\partial \lambda_{ij,st}} = f_{ij} - f_{st}$ and $\frac{\partial D}{\partial \mu_{ij}} = f_{ij} - \log p_{ij} - \sum_{(s,t)\in\mathfrak{B}^*(i,j)} \log(1-p_{st})$.

We have the following theorem.

Theorem 4: Let $(\mathbf{p}(n), \mathbf{f}(n), \boldsymbol{\mu}(n), \boldsymbol{\lambda}(n))$ denote the sequence of vectors defined by the iterative procedure stated in (31)-(34) when the step size is $\gamma_n = \gamma, \forall n$. Then there exists a $\bar{\gamma}\in\mathfrak{R}^+$ such that when $\gamma < \bar{\gamma}$, every accumulation point $(\mathbf{p}^*, \mathbf{f}^*, \boldsymbol{\mu}^*, \boldsymbol{\lambda}^*)$ of the sequence $(\mathbf{p}(n), \mathbf{f}(n), \boldsymbol{\mu}(n), \boldsymbol{\lambda}(n))$ is primal-dual optimal.

TABLE III

THE TOTAL THROUGHPUT ($x_A + x_B + x_C$) AND FAIRNESS INDEX ($r = \frac{x_A}{x_B}$) UNDER DIFFERENT SCHEDULING METHODS

Scheduling Methods	Proportional Fairness	Two-Hop Fairness	Max-Min Fairness
Total Throughput	1.0000	0.7693	0.7959
Fairness Index	4.0000	0.7501	1.0000

3) Achieving Lexicographic Fair Rate Allocation for the Rate Approximations: Another widely used definition of max-min fairness [13] is one in which the link rates are maximized in a lexicographic sense. In the lexicographic max-min fair allocation, any link rate x_e cannot be increased without decreasing some other link rate $x_{e'}$ which is smaller than or equal to x_e. Although this is in general a multi-criteria optimization problem, our approaches achieve the lexicographic max-min fairness for the rate approximation in the CSMA/CA network.

Lemma 1: When the max-min rate is achieved for v_{ij} in the CSMA/CA network, the links that belong to the same strongly connected components in the link graph have the same rate approximation v_{ij}.

We have shown that it is a reasonable assumption that the link graph for the CSMA/CA network is connected. Since the link graph for the rate approximation is bidirectional, it is strongly connected. From *Lemma 1*, we conclude that when the max-min fairness is achieved for the rate approximation, all the links have the same v_{ij} value. Therefore the proposed algorithm can be used to attain lexicographic max-min fairness for the rate approximations.

D. Simulation Investigation

In this section, we investigate the performance of the three distributed scheduling methods with simulations of the three-link scenario.

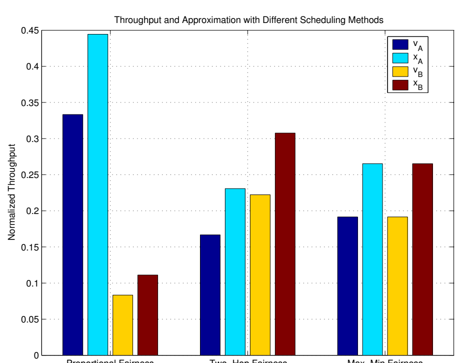

Fig. 10. Throughput and approximation values under different scheduling methods.

The approximation and the throughput achieved by these three methods are plotted in Fig. 10, and the total throughput achieved in the network and the fairness index (the ratio of throughput on link A to that on link B) are presented in Table III.

From the simulation results, it can be seen that proportional fairness achieves the largest total throughput. But it also provides the worst fairness amongst the three links. For the two-hop fairness method, it is much better than the proportional fairness in terms of fairness. However the total throughput achieved is also the lowest. This is due to the fact that link B, whose activity will block the other two links, is assigned the highest scheduling rate. The max-min fairness scheduling is the best trade-off between total throughput and the fairness. However, since it is implemented in an iterative way, it has higher complexity than the other two scheduling methods.

VI. CONCLUSIONS

We address the throughput modelling and fairness provision for CSMA/CA networks in this paper. A throughput model based on Markov analysis is proposed and its performance is verified with simulations. Fairness issues are discussed based on this model. Furthermore, we proposed the throughput approximations based on local topology information and suggested distributed scheduling methods to provide different fairness guarantees in the network.

APPENDIX I
PROOF OF THEOREM 1

Proof: For each subset $D \subset A \cup B$, we define $D_A = D \cap A$ and $D_B = D \cap B$. It follows that D_A and D_B satisfy

$$D = D_A \cup D_B \quad D_A \subset A \quad D_B \subset B$$

Noting that D_A is the set of links that can be carried out simultaneously, $\Psi(A)$ has the item $\prod_{i \in D_A} \rho_i$. Similarly $\Psi(B)$ has the item $\prod_{i \in D_B} \rho_i$. So $\Psi(A)\Psi(B)$ has the item $\prod_{i \in D_A \cup D_B} \rho_i$, i.e. the item $\prod_{i \in D} \rho_i$. Therefore each item of $\Psi(A \cup B)$ is also an item in $\Psi(A)\Psi(B)$.

On the other hand, let D_A be a subset of links of A that can be scheduled simultaneously and D_B denotes a subset of links of B that can be scheduled simultaneously, then $\Psi(A)$ has the item $\prod_{i \in D_A} \rho_i$ and $\Psi(B)$ has the item $\prod_{i \in D_B} \rho_i$. Hence $\Psi(A)\Psi(B)$ has the item $\prod_{i \in D_A \cup D_B} \rho_i$. If, however, some links in D_A cannot be scheduled with the links in D_B, then surely $D_A \cup D_B$ is not a set of links that can be carried out at the same time, i.e. $\Psi(A \cup B)$ does not have the item $\prod_{i \in D_A \cup D_B} \rho_i$. Therefore, $\Psi(A)\Psi(B)$ may have the item that does not belong to $\Psi(A \cup B)$.

Since each item is positive, and each item of $\Psi(A \cup B)$ is also an item in $\Psi(A)\Psi(B)$ while some items in $\Psi(A)\Psi(B)$ might not be items of $\Psi(A \cup B)$, we can conclude that $\Psi(A \cup B) \leq \Psi(A)\Psi(B)$. We can also know that $\Psi(A \cup B) = \Psi(A)\Psi(B)$ when any set of links in B can be scheduled with a set of links in A.

∎

APPENDIX II
PROOF OF COROLLARY 1

Proof: According to the throughput formula of x_{ij} and the property stated in *Theorem 1*, we have

$$\begin{aligned}
\frac{x_{ij}}{\rho_{ij}} &= \frac{\Psi(\mathfrak{B}(i,j)^c)}{\Psi(\mathfrak{S})} = \frac{\Psi(\mathfrak{B}(i,j)^c)}{\Psi(\mathfrak{B}(i,j)^c \bigcup \mathfrak{B}(i,j))} \\
&\geq \frac{\Psi(\mathfrak{B}(i,j)^c)}{\Psi(\mathfrak{B}(i,j)^c)\Psi(\mathfrak{B}(i,j))} \\
&= \frac{1}{\Psi(\mathfrak{B}(i,j))}
\end{aligned}$$

Therefore, $x_{ij} \geq \frac{\rho_{ij}}{\Psi(\mathfrak{B}(i,j))} = u_{ij}$. ■

APPENDIX III
PROOF OF COROLLARY 3

Proof: From *Theorem 1*, we know that the difference between $\Psi(\mathfrak{B}(i,j))\Psi(\mathfrak{B}(i,j)^c)$ and $\Psi(\mathfrak{S})$ comes from the items $\prod_{i \in D_A \cup D_B} \rho_i$, where $D_A \subset \mathfrak{B}(i,j)$, $D_B \subset \mathfrak{B}(i,j)^c$, and there exists a set of links $D \neq \phi$ such that $D \subset D_A$ and $D \subset \mathfrak{B}(\mathfrak{B}(i,j)^c)$, i.e. not all links in D_A can be scheduled simultaneously with the links in D_B.

Note that the items $\prod_{i \in D_A \cup D_B} \rho_i$ approaches zero when ρ_i, $i \in D_A \cup D_B$, approaches zero. As $\phi \in \mathfrak{S}$, we have $\Psi(\mathfrak{S}) \geq \Psi(\phi) = 1$. It follows then

$$\lim_{\rho_{st} \to 0, \forall (s,t) \in E} \frac{\Psi(\mathfrak{B}(i,j))\Psi(\mathfrak{B}(i,j)^c)}{\Psi(\mathfrak{S})} = 1 \tag{35}$$

Therefore

$$\begin{aligned} & \lim_{\rho_{st} \to 0, \forall (s,t) \in E} \frac{u_{ij}}{x_{ij}} \\ = & \lim_{\rho_{st} \to 0, \forall (s,t) \in E} \rho_{ij} \frac{\Psi(\mathfrak{B}^c(i,j))}{\Psi(\mathfrak{S})} \cdot \frac{1}{\rho_{ij}} \Psi(\mathfrak{B}(i,j)) \\ = & \lim_{\rho_{st} \to 0, \forall (s,t) \in E} \frac{\Psi(\mathfrak{B}(i,j)^c)\Psi(\mathfrak{B}(i,j))}{\Psi(\mathfrak{S})} = 1 \end{aligned}$$

Similarly we can prove that $\lim_{\rho_{st} \to 0, \forall (s,t) \in E} v_{ij}/x_{ij} = 1$. ■

APPENDIX IV
PROOF OF THEOREM 2

Proof: Let $U(\vec{\rho})$ be the objective function of (22). Then

$$U(\vec{\rho}) = \sum_{(i,j) \in E} \{\log(\rho_{ij}) - |\mathfrak{B}(i,j)| \log(1 + \rho_{ij})\} \tag{36}$$

It can be easily verified that $f(x) = \log(x) - N \log(1 + x)$ is a concave function of x for any N. Therefore $U(\vec{\rho})$ is a concave function of $\vec{\rho}$.

Setting $\frac{\partial U}{\partial \rho_{ij}}|_{\vec{\rho}^*} = 0$ in (36) gives

$$\rho_{ij}^* = \begin{cases} \frac{1}{|\mathfrak{B}(i,j)| - 1} & \text{when } |\mathfrak{B}(i,j)| > 1 \\ +\infty & \text{when } |\mathfrak{B}(i,j)| = 1 \end{cases}$$

As $\mathfrak{B}^*(i,j) = \mathfrak{B}(i,j) \setminus \{(i,j)\}$, we have $|\mathfrak{B}^*(i,j)| = |\mathfrak{B}(i,j)| - 1$, and therefore *Theorem 2* holds. ■

APPENDIX V
PROOF OF THEOREM 3

Proof: Since the logarithmic function is strictly increasing, each constraint in (26) can be equivalently represented as $\log v \leq \log(p_{ij}) - \sum_{(s,t) \in \mathfrak{B}^*(i,j)} \log(1 - p_{st})$, for any $(i,j) \in E$. If we denote $f = \log v$, it follows

$$f - \log(p_{ij}) - \sum_{(s,t) \in \mathfrak{B}^*(i,j)} \log(1 - p_{st}) \leq 0 \tag{37}$$

Note that each constraint in (37) forms a convex set of $(f, \mathbf{p})$. As logarithmic function is monotonically increasing, v is maximized when f is maximized. Therefore (26) can be formulated as the following convex program.

$$\begin{aligned} \max \quad & f \\ \text{subj. to} \quad & f - \log(p_{ij}) - \sum_{(s,t) \in \mathfrak{B}^*(i,j)} \log(1 - p_{st}) \leq 0 \\ & \qquad \forall (i,j) \in E \end{aligned} \tag{38}$$

Since $0 < p_{ij} \prod_{(s,t) \in \mathfrak{B}^*(i,j)} (1 - p_{st}) \leq 1$, it follows $f \leq 0$. Instead of maximizing f, we can attain the same optimal solution if we minimize $\frac{1}{2} f^2$. Therefore (38) can be rewritten as

$$\begin{aligned} \max \quad & \tfrac{1}{2} f^2 \\ \text{subj. to} \quad & f - \log(p_{ij}) - \sum_{(s,t) \in \mathfrak{B}^*(i,j)} \log(1 - p_{st}) \leq 0 \\ & \qquad \forall (i,j) \in E \end{aligned} \tag{39}$$

We introduce f_{ij} for each link (i,j) and add the constraints $f_{ij} = f_{st}$ for all links $(s,t) \in \mathfrak{B}(i,j)$. Note that the link graph is assumed to be connected. Therefore the newly added constraints $f_{ij} = f_{st}$ imply that all f_{ij}, $(i,j) \in E$ have the same value. The optimization problem in (39) can be represented as the following form

$$\begin{aligned} \min \quad & \tfrac{1}{2} \sum_{(i,j) \in E} f_{ij}^2 \\ \text{subj. to} \quad & f_{ij} - \log(p_{ij}) - \sum_{(s,t) \in \mathfrak{B}^*(i,j)} \log(1 - p_{st}) \leq 0 \\ & \qquad \forall (i,j) \in E \\ & f_{ij} = f_{st} \qquad \forall (i,j) \in E, \forall (s,t) \in \mathfrak{B}(i,j) \end{aligned}$$

Therefore (27) is essentially the same optimization problem as (26) and hence its solutions provide max-min fairness for the throughput approximations amongst all links in the CSMA/CA network. ■

APPENDIX VI
PROOF OF LEMMA 1

Proof: According to the definition of the link graph for the CSMA/CA network, v_{ij} can be increased by only decreasing v_{st} while throughput approximations on all other links are not decreased if there is an edge from (i,j) to (s,t) in the link graph (this can be achieved by just choosing a smaller p_{st} while fixing p_{ab} for all $(a,b) \in E \setminus \{(s,t)\}$). If the link graph is connected, it is also strongly connected since the graph is bidirectional.

Assume that not all the links have the same throughput approximations when the max-min fairness for throughput approximations v is achieved. Denote the minimum value of the throughput approximations for all the links as $v_{\min}$, i.e. $v_{\min} = \min\{v_{(i,j)}, (i,j) \in E\}$. Denote the set of links, whose throughput approximations are equal to $v_{\min}$, as $\mathfrak{L}_1$, i.e. $\mathfrak{L}_1 = \{(i,j) : v_{ij} = v_{\min}, (i,j) \in E\}$, and denote $\mathfrak{L}_2 = \{(i,j) : v_{ij} > v_{\min}, (i,j) \in E\}$.

Obviously $\mathfrak{L}_1 \cap \mathfrak{L}_2 = \phi$ and $\mathfrak{L}_1 \cup \mathfrak{L}_2 = E$, and under the assumption neither $\mathfrak{L}_1$ nor $\mathfrak{L}_2$ is a null set.

When $\mathfrak{L}_1$ is not empty, we can find a link $(i,j) \in \mathfrak{L}_1$ and a link $(s,t) \in \mathfrak{L}_2$ such that there is an bidirectional edge from (i,j) to (s,t) in the link graph. As $v_{st} > v_{\min}$, we can increase v_{ij} a little by only decreasing v_{st} such that $v_{ij} > v_{\min}$ while still keeping $v_{st} > v_{\min}$. In this way, link (i, j) is removed from the set $\mathfrak{L}_1$ and all links that were in $\mathfrak{L}_2$ are still in $\mathfrak{L}_2$.

We can repeat the above operation until all the elements are removed from $\mathfrak{L}_1$. Now all the links are in the set $\mathfrak{L}_2$, i.e. $v_{ij} > v_{\min}$ for any $(i, j) \in E$. This contradicts with the assumption that $v_{\min}$ is the max-min value of the throughput approximations. Therefore our assumption is not correct and all the links have the same throughput approximations when the max-min fairness is achieved. ■

REFERENCES

[1] IEEE Computer Society LAN MAN Standards Committee, ed., *IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE Std 802.11-1999, The Institute of Electrical and Electronics Engineers, NY, 1999.

[2] Robert R. Boorstyn, Aaron Kershenbaum, Basil Maglaris, Veli Sahin, "Throughput Analysis in Multihop CSMA Packet Radio Networks", *IEEE Transactions on Communications*, Vol.COM-35, No.3, pp.267-274, March 1987.

[3] Aaron Kershenbaum, Robert R. Boorstyn, Mon-Song Chen, "An Algorithm for Evaluation of Throughput in Multihop Packet Radio Networks with Complex Topologies", *IEEE Journal on Selected Areas in Communications*, Vol.SAC-5, No.6, pp.1003-1012, July 1987.

[4] Fouad A. Tobagi, "Modeling and Performance Analysis of Multihop Packet Radio Networks", *Proceedings of the IEEE*, Vol.75, No.1, pp.135-155, January 1987.

[5] Giuseppe Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function", *IEEE Journal on Selected Areas in Communications*, Vol.18, No.3, pp.535-547, March 2000.

[6] V. Bharghavan, A. Demers, S. Shenker, L. Zhang, "MACAW: A Media Access Protocol for Wireless LANs", in *Proceedings of ACM SIGCOMM*, 1994.

[7] H. Luo, S. Lu, V. Bharghavan, "A New Model for Packet Scheduling in Multihop Wireless Networks", in *Proceedings of ACM MobiCom*, August 2000.

[8] B. Bensaou, Y. Wang, and C. Ko, "Fair Media Access in 802.11 based Wireless Ad-Hoc Networks", in *IEEE/ACM MobiHOC*, Boston, MA., August 2000.

[9] Xiao Long Huang, Brahim Bensaou, "On Max-min Fairness and Scheduling in Wireless Ad-Hoc Networks: Analytical Framework and Implementation", in *Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing*, Long Beach, CA, October 2001.

[10] Xin Wang, Koushik Kar, "Max-Min Fair Rate Allocation in ALOHA Networks", in *Proceedings of the* 42^{nd} *Annual Allerton Conference on Communications, Control and Computing*, Urbana-Champaign, USA, October 2004.

[11] Xin Wang, Koushik Kar, "Throughput Modelling and Fairness Issues in CSMA/CA Based Ad-Hoc Networks", *Technical Report*, Department of Electrical Computer and System Engineering, Rensselaer Polytechnic Institute, 2004. Available at http://www.ecse.rpi.edu/∼koushik/csma-fairness-techrep.pdf

[12] Xin Wang, Koushik Kar, "Throughput Modelling and Fairness Issues in CSMA/CA Based Ad-Hoc Networks", To appear in *Proceedings of Infocom 2005*, Miami, March 2005.

[13] Dimitri P. Bertsekas, *Data networks*, Englewood Cliffs, N.J.: Prentice Hall, 1992.

[14] D. P. Bertsekas, *Nonlinear Programming*, Athena Scientific, 1999.