This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

On the Efficiency of CSMA-CA Scheduling in Wireless Multihop Networks

Apoorva Jindal, *Member, IEEE*, and Konstantinos Psounis, *Senior Member, IEEE*

***Abstract*—This paper establishes that random access scheduling schemes, and more specifically CSMA-CA, yield exceptionally good performance in the context of wireless multihop networks. While it is believed that CSMA-CA performs significantly worse than optimal, this belief is usually based on experiments that use rate allocation mechanisms that grossly underutilize the available capacity that random access provides. To establish our thesis, we first compare the achievable rate region of CSMA-CA and optimal in a number of carefully constructed multihop topologies and find that CSMA-CA is always within 48% of the optimal. Motivated by this result, we next characterize the worst-case performance of CSMA-CA in neighborhood topologies representing the congested regions of larger multihop topologies by deriving the neighborhood topology that yields the worst-case throughput ratio for CSMA-CA and find that in neighborhood topologies with less than 20 edges: 1) CSMA-CA is never worse than 16% of the optimal when ignoring physical-layer constraints; and 2) in any realistic topology with geometric constraints due to the physical layer, CSMA-CA is never worse than 30% of the optimal. Considering that maximal scheduling achieves much lower bounds than the above, and greedy maximal scheduling, which is one of the best known distributed approximation of an optimal scheduler, achieves similar worst-case bounds, CSMA-CA is surprisingly efficient.**

***Index Terms*—CSMA-CA, optimality, wireless multihop networks.**

I. Introduction

A FUNDAMENTAL open problem in the design of wireless multihop networks is to efficiently schedule transmissions in a distributed manner. Implementing a scheduling scheme that supports the largest capacity region (referred to as optimal scheduling in this paper) involves solving a max-weight problem with secondary interference constraints, which is provably NP-hard [30]. Thus, researchers have focussed on proposing efficient approximation algorithms like maximal and greedy maximal scheduling; see [8], [16], [21], [29], and references therein.

Manuscript received December 14, 2009; revised October 18, 2010, December 03, 2011, and August 19, 2012; accepted October 05, 2012; approved by IEEE/ACM Transactions on Networking Editor S. Shakkottai. The work of K. Psounis was supported by the Army Research Laboratory (ARL) Collaborative Technology Alliance (CTA) under Grant W911NF-09-2-0053, Marie Curie Grant 256416, a Cisco Systems Research Center (CRC) grant, and the National Science Foundation (NSF) Networking Technology and Systems (NeTS) under Grant 0832161.

A. Jindal is with Juniper Networks, Inc., San Jose, CA 95134 USA (e-mail: ajindal@juniper.net).

K. Psounis is with the Department of Electrical Engineering, University of Southern California, Los Angeles, CA 90089 USA (e-mail: kpsounis@usc.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNET.2012.2225843

However, it is not clear if these approximation algorithms can be implemented in a distributed manner, as they tend to have a high control overhead in terms of control messages exchanged per transmission. To reduce overhead, researchers have investigated implementing these deterministic scheduling schemes using random access. For example, [12], [17], and [34] propose random access schemes to approximate greedy maximal scheduling. Furthermore, CSMA-CA can be viewed as a mechanism to get maximal schedules [8], hence it can be considered a random access realization of maximal scheduling. The underlying assumption in this approach is that the random access schemes are easy to implement, but suffer from a performance degradation due to suboptimal decisions and collisions.

Using as an example CSMA-CA, which is the scheduling algorithm used in the *de facto* protocol IEEE 802.11, the main contribution of this paper is to demonstrate that a well-designed random access scheme yields a throughput performance close to the optimal and outperforms equivalent deterministic approximation algorithms. Thus, random access schemes like the ones currently used in deployed wireless networks are not only easy to implement, but, contrary to popular belief, also yield superior throughput performance. Intuitively, well-designed random access schemes avoid very bad scheduling decisions thanks to carrier sense and collision avoidance ideas. Furthermore, they choose among the rest of the schedules randomly, which benefits from the well-known performance traits of randomized algorithms [22]. Thus, the end result turns out to be quite good.

A. Contributions

Questions about the suitability of random access scheduling schemes for multihop networks arise from prior works, which observed unfair and inefficient throughputs when IEEE 802.11 is used with TCP or with no rate control (all sources pump in data as fast as possible) [11], [19], [24], [28], [32], [33], [36]. However, no one has studied how IEEE 802.11 does with optimal rate control.[1] To motivate an analytical study of the worst-case performance of random access schemes in multihop topologies, we first evaluate CSMA-CA with optimal rate control for a number of carefully constructed multihop topologies (in which we include the most common topologies used by prior works to show that IEEE 802.11 does not work well with TCP and with no rate control) and observe that CSMA-CA is always within 48% of the optimal for all the topologies we study. This study also yields certain nonintuitive observations on which interference characteristics degrade/improve CSMA-CA's throughput performance.

[1]Optimal rate control for a particular scheduling scheme is defined to be the rate control algorithm that achieves capacity over that scheduling algorithm. Each scheduling scheme may have a different optimal rate control algorithm, however, what concerns us is that it is capacity-achieving.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

To explain these observations, we derive the one-hop *neighborhood topology* (this topology represents the congested region of a larger multihop topology and is defined in Section III) that yields the worst-case throughput ratio for CSMA-CA. We also compare this ratio to the worst-case throughput ratio of maximal and optimal scheduling derived in [8]. CSMA-CA is a random access algorithm that also yields maximal schedules, where a maximal schedule is one where no more edges can be scheduled due to interference constraints. However, unlike maximal scheduling, it does not yield collision-free schedules, and simultaneous transmissions may collide due to suboptimal scheduling. This leads to a loss in throughput. Thus, one may expect that CSMA-CA will have a worse performance, which, nevertheless, turns out not to be the case.

In particular, in topologies with neighborhoods not larger than 20 edges, for a max-min fair rate allocation, we establish the following through numerical simulation: 1) CSMA-CA achieves more than 16% of the optimal throughput always; 2) in any real system where the physical-layer model imposes geometrical constraints on the topology, CSMA-CA achieves more than 30% of the optimal throughput. To compare, under the same assumptions, maximal scheduling achieves 5% and 12.5%, respectively.

The fact that CSMA-CA achieves more than 30% of the optimal throughput in real systems may not seem that impressive at first glance. However, note that implementing optimal scheduling is NP-hard, and CSMA-CA is a distributed scheduling algorithm that uses only two control messages per packet exchange. To get a flavor of the state of the art, note that greedy maximal scheduling, which is one of the best known deterministic approximate scheduling schemes, has a lower bound of 1/4 on its throughput efficiency [18], which is comparable to what CSMA-CA achieves.

While characterizing the worst-case neighborhood topology, surprisingly we find that the less the interference is in the topology, the worse CSMA-CA's throughput ratio is because CSMA-CA schedules noninterfering edges independently while optimal scheduling will schedule them simultaneously. Hence, the performance gain for optimal scheduling with decreasing interference is larger than what it is for CSMA-CA.

B. Related Work

As stated earlier, most of the related prior works study the performance of IEEE 802.11 with either TCP or with no rate control at the source in a multihop network; see, for example, [11], [19], [24], [28], [32], [33], [36], and references therein. A few papers have looked at deriving performance bounds for different random access schemes. Reference [7] derived the upper bound on the throughput achieved with any asynchronous random access scheme with physical carrier sensing. We assume a more sophisticated random access protocol, namely CSMA-CA, which also uses virtual carrier sensing, exponential backoffs, and exchange of RTS-CTS messages, and hence get a higher bound. References [14] and [25] proposed new random access schemes and proved their throughput optimality. However, to simplify analysis, their analysis neglected either the effect of collisions in multihop networks [14] or assumed synchronous transmissions [25]. Both these papers also neglect the effect of topology asymmetries [11], [15] in multihop networks.

The good performance of CSMA-CA has been observed by other papers. To our knowledge, [19] is the first paper to observe that CSMA-CA does well for a chain topology with optimal rate control. However, the main focus of that paper was to study CSMA-CA with no rate control, and it did not go forward with this observation. In one of our prior works [15] where we presented an analytical method to compute the achievable rate region of IEEE 802.11 in multihop topologies, we also observed via simulations that in the few topologies we studied, CSMA-CA was within 64% of the optimal. However, the paper does not investigate this further neither does it offer a formal explanation for this observation. Finally, a recent paper [5] shows the following for a *two-sender case*: Carrier sense performance is surprisingly close to optimal, and hidden and exposed terminal phenomena cause only a mild reduction in throughput in a two-edge topology.

II. CSMA-CA With Optimal Rate Control

This section presents a simulation-based study of carefully constructed multihop topologies with optimal rate control over CSMA-CA. The objective is to characterize the end-to-end performance of CSMA-CA in multihop topologies and determine which interference characteristics degrade/improve CSMA-CA's throughput performance. In this section, we will merely state these results. Explaining the cause of these results will be accomplished in Section III.

To evaluate CSMA-CA with optimal rate control, and contrast this evaluation with TCP over CSMA-CA or with no rate control over CSMA-CA, we compare the following four results: 1) the achievable rate region with optimal scheduling; 2) the achievable rate region with CSMA-CA; 3) the rates achieved by TCP over CSMA-CA; and 4) throughputs achieved assuming that all sources always have a packet to send (no rate control) over CSMA-CA. This is referred to as the saturation throughput [4] and has been studied in detail for CSMA-CA in multihop networks [11], [20].

A. Methodology

1) Deriving Each Result: We first describe the procedure adopted to derive each result. Note that a transport protocol like TCP also guarantees reliability. The destination sends back an ACK packet for each received DATA packet. Since the ACK packets will interfere with the DATA packets, their presence will reduce the achievable rate region. For a fair comparison, we will incorporate the effect of ACK packets while deriving all four results. The size of the ACK packets is set to the size of ACK packets in TCP.

1) Achievable rate region with optimal scheduling: We use the methodology proposed by Jain *et al.* [13]. For a fair comparison, we assume that the overhead imposed by the control message exchange and protocol headers is equal for IEEE 802.11 (the protocol that implements CSMA-CA scheduling) and optimal scheduling.[2] The control messages exchanged for IEEE

[2]We expect the actual overhead required to achieve optimal scheduling to be much higher. Hence, the comparison is geared to favor optimal scheduling which makes the obtained good results for CSMA-CA even stronger.

802.11 scheduling are: RTS, CTS and 802.11-ACK.[3] The protocol headers included in the DATA packet are the physical-layer, MAC-layer, network-layer, and transport-layer headers. We measure the bandwidth consumed by these messages and headers and factor it in the calculations for optimal scheduling. This ensures that the loss in throughput with CSMA-CA is entirely due to inefficiency in scheduling and random backoffs.

2) Achievable rate region with CSMA-CA: We simulate sources generating DATA packets and destinations generating ACK packets at a constant bit rate. UDP is used at the transport layer. To derive the entire region, we simulate all possible combinations of flow rates with each flow rate varying in steps of 10 kb/s and checking if the input rates were achievable.
3) Performance of TCP over CSMA-CA: We simulate sources generating FTP traffic and use TCP at the transport layer to deliver packets. We use TCP SACK, but with Nagle's algorithm [23] and the delayed ACK mechanism [9] turned off, in our simulations.
4) Saturation throughput: We simulate two CBR flows (DATA from source to the destination and ACK from the destination to the source) for each source–destination pair. Rates of both the flows are set to 100 Mb/s so that the queues at both the source and the destination are always full.

2) Simulation Setup: To generate simulation results with CSMA-CA scheduling, we use Qualnet 4.0 [2] as the simulation platform. All our simulations are conducted using an unmodified IEEE 802.11(b) MAC (DCF) with RTS/CTS. We use the default parameters of IEEE 802.11(b) in Qualnet [2] unless otherwise stated. To study CSMA-CA scheduling only, the other features of IEEE 802.11 protocol like the auto-rate adaptation feature at the PHY layer are turned off, and simulations are conducted for the following two data rates, 1 and 11 Mb/s, which are the minimum and the maximum data rates allowed by IEEE 802.11(b). All our simulations are conducted with zero channel losses, and we set the buffer size and maximum retry limit in IEEE 802.11 (the number of retransmission attempts after which the packet is dropped) to a very large value to avoid packet losses. This allows us to generate the achievable rate region without having to worry about transport-layer retransmissions to recover from these losses. The packet size is fixed to be 1024 B.

3) Max-Min Fair Rate Allocation: A feasible allocation of rates $\vec{x}$ is max-min fair if and only if an increase of any rate within the domain of feasible allocations must be at the cost of a decrease of some already smaller rate. Formally, for any other feasible allocation $\vec{y}$, if $y_f > x_f$, $f \in F$, then there must exist some $f' \in F$ such that $x_{f'} \leq x_f$ and $y_{f'} < x_{f'}$ [3].

Unless explicitly stated, if there are more than five flows in the network, we do not calculate the entire achievable rate region for CSMA-CA because simulating every possible combination of flow rates to determine the achievable rate region does not scale and, in addition, the corresponding multidimensional region would not be very instructive anyway. Instead, we will compare the max-min fair rate allocation with CSMA-CA and optimal scheduling because it can be derived through simulations using the algorithm described in [3] to determine max-min allocations in wired networks, except that now any flow passing through the neighborhood of the edge that gets congested is throttled. This modification is required for wireless networks because a flow passing through the neighborhood of an edge will eat up capacity at that edge due to interference.

4) Comparing CSMA-CA and Optimal Scheduling: To compare CSMA-CA and optimal scheduling, we compare the sum throughput at the max-min allocation with CSMA-CA and optimal scheduling. Specifically, if $x_f^{\text{CSMA-CA}}$ and x_f^{OPT} denote the rate of flow $f \in F$ at the max-min rate allocation with CSMA-CA and optimal scheduling, respectively, then we say that CSMA-CA is within $x\%$ of the optimal if $x = 100 \sum_{f \in F} x_f^{\text{CSMA-CA}} / \sum_{f \in F} x_f^{\text{OPT}}$.

To compare the throughput at the congested edge, we have to pick up a rate allocation point to compare. Among the commonly used rate allocation points, like proportionally fair rate allocation, maximum sum throughput rate allocation, and max-min rate allocation, we choose the max-min rate point because of the following two reasons: 1) It yields the worst throughput ratio at edge e_c among the three.[4] Intuitively, more collisions are observed at the max-min allocation, which results in a lower throughput with CSMA-CA. 2) The max-min fair rate point can be determined without having to find the entire achievable rate region. This allows us to compare CSMA-CA and optimal scheduling for larger topologies with several flows.

Note that a methodology similar to the one presented in this paper can be used to compare the throughput of the congested edge at any other rate allocation point.

5) Congested Neighborhoods: We formally defined congested neighborhoods in our prior work [28]. For completeness, we reproduce the definition here. We then describe the methodology used to determine the congested neighborhoods in a topology and then comment on their significance.

Let Tx_e and Rx_e denote the transmitter and the receiver of an edge e. Two edges e_1, e_2 are said to interfere with each other if either Tx_{e_1} interferes with either Tx_{e_2} or Rx_{e_2}, or Rx_{e_1} interferes with either Tx_{e_2} or Rx_{e_2}. A neighborhood of an edge e is defined to be the set of edges that interfere with e. At the max-min allocation, the edges whose queues have a utilization equal to one (in other words, the arrival rate is equal to the service rate) are defined to be congested edges, and the neighborhood of congested edges is defined to be congested neighborhoods.

To determine the congested neighborhood, we first determine the max-min allocation using the methodology described in Section II-A.3. We then simulate the topology with CBR flows and fixing the rate of each end-to-end flow to be equal to its corresponding rate in the max-min allocation. To account for simulation errors, any edge whose queue utilization is greater than 0.95 is labeled a congested edge, and its neighborhood is marked as a congested neighborhood. For each topology, the congested edge will be denoted by a symbol depicting a queue.

[3]We call the ACK messages exchanged by the IEEE 802.11 protocol as 802.11-ACK to distinguish it from the transport-layer ACKs.

[4]We verified this statement for all the multihop topologies we study in this paper, except for the one depicted in Fig. 3(m). This topology has 30 flows, hence it is not possible to determine the other rate allocations.

TABLE I
RESULTS FOR A SINGLE-EDGE TOPOLOGY

Experiment Description	Throughput with 1 Mbps data rate	Throughput with 11 Mbps data rate
Achivable rate with optimal scheduling	0.695 Mbps	3.23
Achievable rate with CSMA-CA	0.67 Mbps	2.76 Mbps
TCP over CSMA-CA	0.66 Mbps	2.62 Mbps
Saturation Throughput	0.67 Mbps	2.76 Mbps

We discuss the significance of congested neighborhoods in detail in Section III-A. The underlying idea is that the throughput performance of a given multihop topology at the max-min rate allocation is dictated by the throughputs achieved in the congested neighborhoods.

B. *CSMA and Optimal Scheduling*

In this section, we compare the achievable rate region with CSMA-CA and optimal scheduling for several different multihop topologies. We divide the topologies into three groups. For each group, we first describe the characteristics of the topologies belonging to that group, and then present results for specific topologies.

Before studying multihop topologies, we first present the throughput results on an one-edge topology with one transmitter and one receiver and no interfering edge in Table I. This serves as a baseline. We make the following observations from these results.

1) At 1 Mb/s, CSMA-CA is within 96% of the optimal, while at 11 Mb/s, it is within 85%. We first discuss why CSMA-CA achieves a lower throughput with 11 Mb/s data rate. Before each packet transmission, the transmitter backoffs (waits) for a randomly selected duration. The expected value of this random duration is a constant irrespective of the data rate. Thus, a fraction of available throughput is lost due to backoffs. We refer to this loss in throughput as the random access overhead. Now, the lower the packet transmission time is, the more the random access overhead, which explains the extra loss in throughput at the higher data rate.
2) Even though the data rate is 11 Mb/s, even with optimal scheduling, one can achieve a throughput of only 3.23 Mb/s. Recall that the control overhead of optimal scheduling is assumed to be the same as that of CSMA-CA. Thus, more than 70% of the available throughput is consumed by control overhead (which includes MAC control packets and protocol headers). As discussed in detail in Section II-C, this significant control overhead is an artifact of auto-rate adaptation and has nothing to do with CSMA-CA.

1) Two-Edge Topologies: The following four two-edge topologies [11], [15] represent the fundamental ways in which edges interact with each other: 1) *Coordinated Stations* [Fig. 1(a)]; 2) *Near Hidden Edges* [Fig. 1(b)]; 3) *Asymmetric Topology* [Fig. 1(c)]; 4) *Far Hidden Edges* [Fig. 1(d)].

Since our prior work [15] has also studied the achievable rate region of these four two-edge topologies, to avoid repetition, we do not present these regions here. Instead, we merely summarize

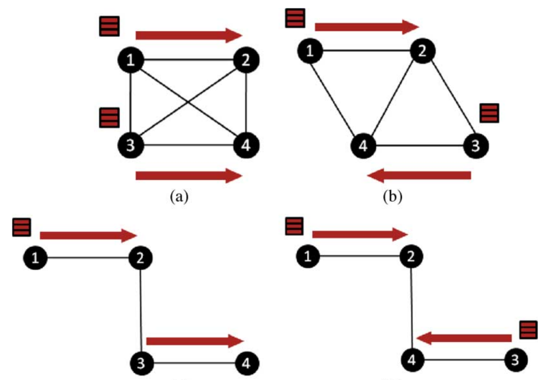

Fig. 1. (a) Coordinated Stations. (b) Near Hidden. (c) Asymmetric. (d) Far Hidden.

TABLE II
RESULTS FOR TWO-EDGE TOPOLOGIES

Topology	Throughput ratio (1 Mbps)	Throughput ratio (11 Mbps)
Coordinated Stations	97.1%	88.8%
Near Hidden Edge	94.2%	81.4%
Asymmetric Topology	80.7%	74%
Far Hidden Edges	86.8%	78.7%

the throughput ratio of CSMA-CA and optimal scheduling at the max-min rate allocation in Table II. *We observe that among the four two-edge topologies, CSMA-CA suffers the maximum throughput loss in the asymmetric topology*.

2) Flow in the Middle and Variants: In topologies belonging to this category, each flow experiences a different level of interference. For example, in the topology presented in Fig. 2(a), the middle flow interferes with the two outer flows, while the outer flows do not interfere with each other; hence, the middle flow experiences more interference than the outer two flows. With TCP and with no rate control (saturation) over CSMA-CA, the flow that experiences a higher level of interference experiences unfair throughputs, and even starvation. Hence, these topologies have been extensively studied in the literature by works that focus on the unfairness and starvation issues with TCP/saturation over CSMA-CA [11], [28], [33]. Therefore, to avoid repetition, we will not discuss the throughput results with TCP and saturation for these set of topologies. Instead, we focus on the throughputs achievable with optimal rate control.

Flow in the Middle: Fig. 2(a) depicts the topology. There is only one congested edge: $3 \rightarrow 4$. All three flows contribute to congestion on this edge. The middle flow $3 \rightarrow 4$ interferes with both the outer flows, while the outer flows do not interfere with each other. Fig. 2(b) and (c) compares the achievable rate regions at 1- and 11-Mb/s data rates, respectively. (By symmetry, the outer two flows will achieve approximately the same rate for any scheme.) At the max-min allocation, CSMA-CA is within 72.6% and 62.5% of the optimal, respectively. *Thus, CSMA-CA with a well-designed rate control protocol will yield fair and efficient throughputs for this topology*.

Stack: Fig. 2(d) depicts the topology. It is similar to the flow in the middle topology, except now each flow goes through two hops instead of one. Also, only the inner nodes—2, 5, and 8—interfere with each other (2 with 5, and 5 with 8; see figure). There

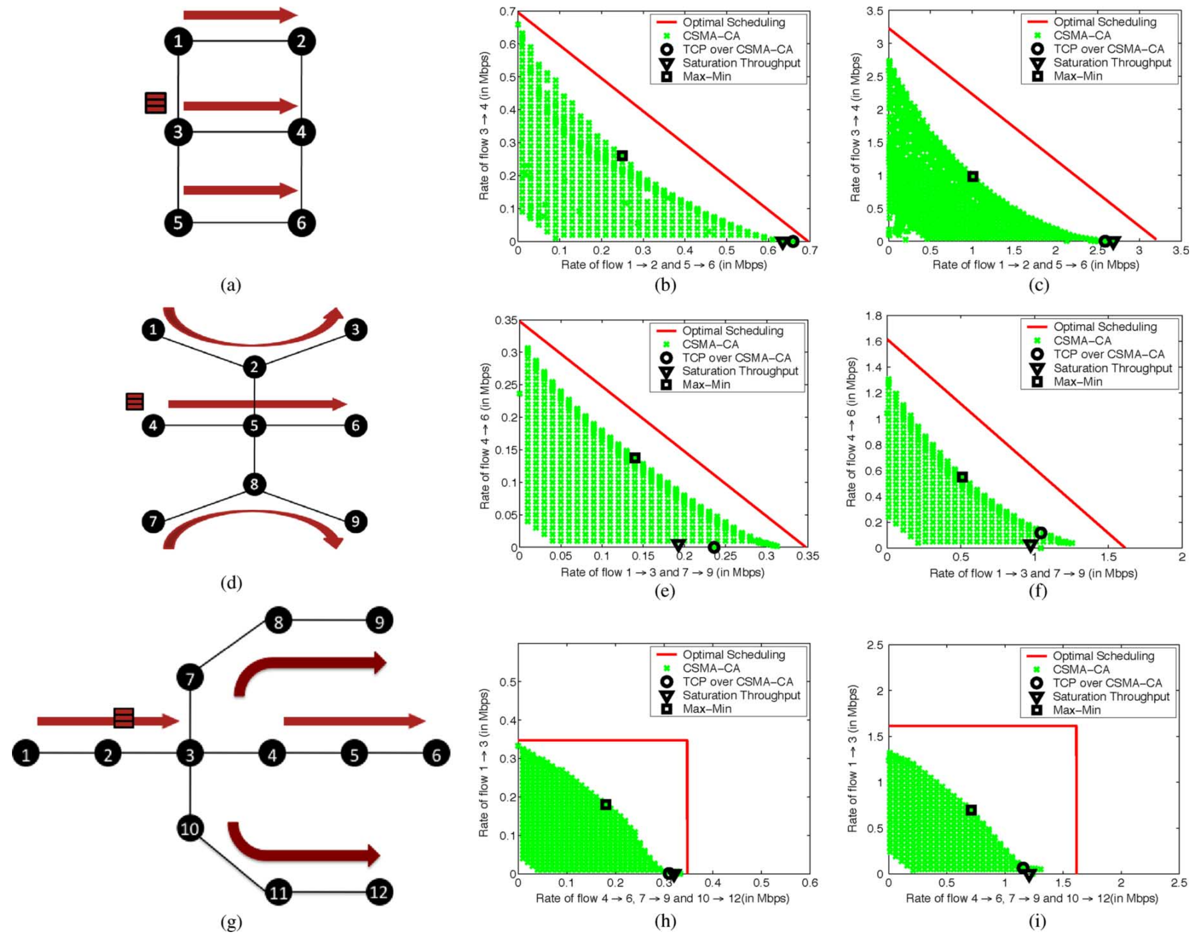

Fig. 2. *Flow in the Middle*: (a) Topology. Achievable rate region. (b) Data rate = 1 Mb/s. (c) Data rate = 11 Mb/s. *Stack*: (d) Topology. Achievable rate region. (e) Data rate = 1 Mb/s. (f) Data rate = 11 Mb/s. *Fork*: (g) Topology. Achievable rate region. (f) Data rate = 1 Mb/s. (i) Data rate = 11 Mb/s.

is still only one congested edge $4 \rightarrow 5$. Fig. 2(e) and (f) compares the achievable rate regions at 1- and 11-Mb/s data rates, respectively. (By symmetry, the outer two flows will achieve approximately the same rate for any scheme.) At the max-min allocation, CSMA-CA is within 79.9% and 64.7% of the optimal, respectively. *Thus, there is not much change in the performance of CSMA-CA by reducing interference at the outer nodes*.

Fork: Fig. 2(g) depicts the topology. It is similar to the flow in the middle topology, except that now the middle flow interferes with three noninterfering flows instead of just two. Fig. 2(h) and 2(i) compares the achievable rate regions at 1- and 11-Mb/s data rates, respectively. (By symmetry, the three noninterfering flows will achieve approximately the same rate for any scheme.) At the max-min allocation, CSMA-CA is within 56.1% and 48.3% of the optimal, respectively. There is an additional loss in throughput over the flow in the middle topology. *Hence, we observe that the greater the number is of noninterfering flows that interfere with the middle flow, the worse is the performance of CSMA-CA*.

3) Chain and Variants: In topologies belonging to this category, there is at least one flow that goes over multiple hops. With TCP and with no rate control over CSMA-CA, the throughput achieved for the flow that goes over multiple hops is very inefficient. Hence, these topologies have been extensively studied in the literature that focus on the inefficiencies in throughput with TCP/saturation over CSMA-CA [19], [24], [32]. However, with optimal rate control, we observe that CSMA-CA allocates rates within 48% of the optimal for the topologies belonging to this category.

Chain: Fig. 3(a) depicts the topology. It has two long flows, $1 \rightarrow n$ and $n \rightarrow 1$. Fig. 3(b) and (c) compares the achievable rate regions at 1- and 11-Mb/s data rates, respectively, for $n = 15$. We make the following observations. 1) At the max-min allocation, CSMA-CA is within 58.3% and 50.4% of the optimal, respectively. 2) TCP and saturation allocate inefficient rates over CSMA-CA. 3) *We performed additional simulations for different values of* n, $6 \leq n \leq 15$, *and observed that changing the number of hops does not change the performance of CSMA-CA with optimal rate control*.

Chain with Two Interfering Short Flows: We next study a chain topology with one long flow that interferes with multiple short flows. Fig. 3(d) depicts the topology. It has one long flow and two short flows $2 \rightarrow 8$ and $\rightarrow 9$, both of which interfere with each other as well as the long flow. Fig. 3(e) and (f) compares the achievable rate regions at 1- and 11-Mb/s data rates, respectively. (By symmetry, the two short flows will achieve approximately the same rate for any scheme.) We make the following observations. 1) At the max-min allocation, CSMA-CA is within 82.5% and 63.9% of the optimal, respectively. *Thus, contrary to intuition, increasing the interference*

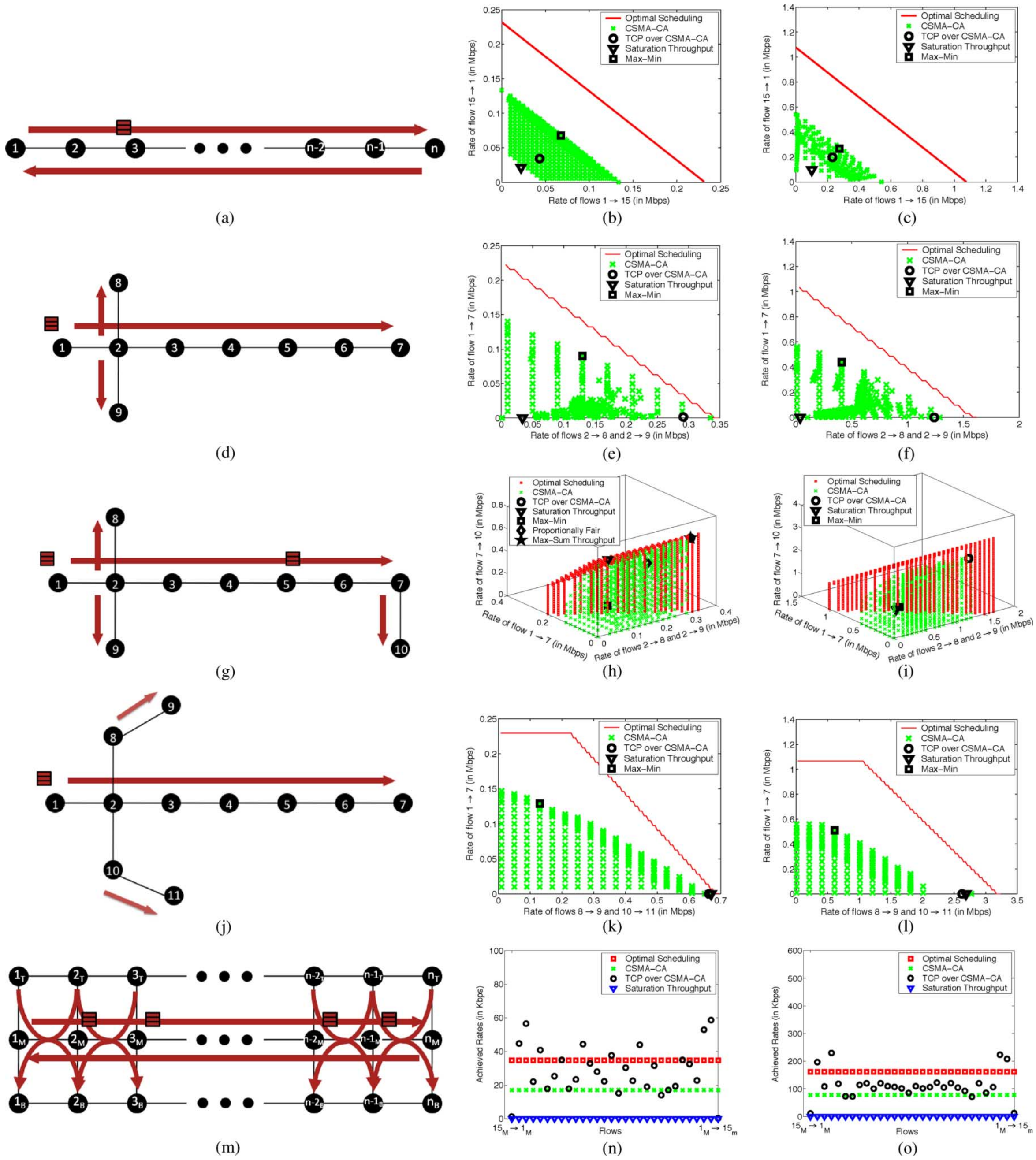

Fig. 3. *Chain*: (a) Topology. Achievable rate region. (b) Data rate = 1 Mb/s. (c) Data rate = 11 Mb/s. *Chain with Two Interfering Short Flows*: (d) Topology. Achievable rate region. (e) Data rate = 1 Mb/s. (f) Data rate = 11 Mb/s. *Chain with Three Interfering Short Flows*: (g) Topology. Achievable rate region. (h) Data rate = 1 Mb/s. (i) Data rate = 11 Mb/s. *Chain-Cross*: (j) Topology. Achievable rate region. (k) Data rate = 1 Mb/s. (l) Data rate = 11 Mb/s. *Parking-Lot*: (m) Topology. Max-min fair rate allocation. (n) Data rate = 1 Mb/s. (o) Data rate = 11 Mb/s.

around the congested edge improves the throughput ratio. 2) TCP and saturation starve the long flow. Actually, since the ACK packets for the two short flows go over $8 \rightarrow 2$ and $9 \rightarrow 2$, which forms hidden terminals with edge $3 \rightarrow 4$ [11], saturation yields near-zero throughputs for these short flows as well.

Chain With Three Interfering Short Flows: Fig. 3(g) depicts the topology. It is similar to the topology depicted in Fig. 3(d) except for the additional flow $7 \rightarrow 10$. Now there are two congested neighborhoods in the topology, around edges $1 \rightarrow 2$ and $5 \rightarrow 6$, and the long flow passes through both. Fig. 3(h) and (i) compares the achievable rate regions at 1- and 11-Mb/s data rates, respectively. (By symmetry, the two short flows $2 \rightarrow 8$ and $2 \rightarrow 9$ will achieve approximately the same rate for any scheme.) We observe that the rate region for CSMA-CA decreases as compared to the chain with two interfering short flows, and now, at the max-min allocation, it is within 57.2% and 545% of the optimal, respectively. *Thus, we observe that the presence of the second congested neighborhood degrades the performance of CSMA-CA as compared to the chain topology with two interfering short flows, which has only one congested neighborhood*. We also note that the actual throughput value of flows $1 \rightarrow 7$, $2 \rightarrow 8$, and $2 \rightarrow 9$ is the same for both the chain with two interfering short flows and the chain with three interfering short flows. Its the lower value of throughput achieved for flow $7 \rightarrow 10$ with CSMA-CA as compared to optimal scheduling, which reduces the net ratio for the chain with three interfering short flows.

Chain-Cross: Fig. 3(j) depicts the topology. It is similar to the topology depicted in Fig. 3(d), except now the short flows around node 2 ($8 \rightarrow 9$ and $10 \rightarrow 11$) do not interfere with each other. Fig. 3(k) and (l) compares the max-min rate allocation at 1- and 11-Mb/s data rates, respectively. (By symmetry, the two short flows $8 \rightarrow 9$ and $10 \rightarrow 11$ will achieve approximately the same rate for any scheme.) At the max-min allocation, CSMA-CA is within 56.5% and 54.1% of the optimal, respectively. Thus, with flows around node 2 ($8 \rightarrow 9$ and $10 \rightarrow 11$) not interfering with each other, the performance of CSMA-CA deteriorates as compared to the topology depicted in Fig. 3(d). *Therefore, we conclude that the presence of flows that interfere with a common flow but do not interfere with each other degrades CSMA-CA's achievable rate region*.

Parking-Lot: Fig. 3(m) depicts the topology. It has two long flows, one in each direction similar to the chain topology depicted in Fig. 3(a). In addition, it has a number of smaller flows, most of which do not interfere with each other, interfering with the two long flows. Finally, both long flows traverse multiple congested neighborhoods in the topology. Fig. 3(n) and (o) compares the max-min rate allocations[5] at 1- and 11-Mb/s data rates, respectively, for $n = 15$. CSMA-CA is within 48.9% and 48.3% of the optimal, respectively, at the max-min allocation. *Not surprisingly, among all the topologies studied, for this topology CSMA-CA has the worst performance*. Additionally, note that TCP not only completely starves the two long flows, but is also unfair to the intermediate small flows. Finally, saturation yields extremely inefficient rates for this topology.

C. Overheads

Control Overhead of CSMA-CA: As discussed at the start of this section, at 11-Mb/s data rate, more than 70% of the available throughput is consumed by control overhead (which includes MAC control packets and protocol headers). The following observation explains the source of this overhead.

An IEEE 802.11 transmitter can transmit at one of the four available basic data rates [1]. The actual data rate employed depends on the condition of the channel and can change as the channel conditions change. This is called auto-rate adaptation. The PHY-layer header contains information used to determine the data rate of the incoming transmission, and hence is always transmitted at 1 Mb/s [1]. Moreover, the PHY-layer header is exchanged for both control (RTS, CTS, and 802.11-ACK) and DATA packets. For a data rate of 11 Mb/s, the transmission time of the 1024-B DATA packet is comparable to the transmission time of the PHY-layer header that is transmitted at 1 Mb/s. Note that a similar overhead is incurred for the much smaller 40-B transport-layer ACK packets. Thus, the large control overhead is an artifact of the auto-rate adaptation implemented at the PHY layer in the IEEE 802.11 protocol and has nothing to do with the scheduling protocol CSMA-CA.

[5]Recall that we compare only the max-min rate allocation for topologies with more than five flows.

The Random Access Overhead: Before each packet transmission, the transmitter backoffs (waits) for a randomly selected duration. Thus, a fraction of available throughput is lost due to backoffs. Recall that we refer to this loss in throughput as the random access overhead. Now, the lower the packet transmission time, the more is the random access overhead as the expected value of this random duration is a constant irrespective of the data rate.

All topologies we study achieve a lower throughput ratio at 11-Mb/s data rate than at 1-Mb/s data rate. The reason is the larger random access overhead at 11 Mb/s. As observed in [20], choosing a smaller value of W_0 when data rate is equal to 11 Mb/s will reduce this random access overhead. For example, choosing $W_0 = 8$ and retaining the default values for the rest of the IEEE 802.11 parameters compensates the extra loss in throughput at 11-Mb/s data rate. Hence, the throughput ratio comparison at 1 Mb/s yields a better and more fair evaluation of the performance of CSMA-CA for any topology with default 802.11 parameters. Therefore, in the rest of our study, the throughput ratios will be evaluated at 1 Mb/s.

D. Summary

In this section, we summarize the intuitive observations derived regarding which topology characteristics deteriorate or improve CSMA-CA's performance with optimal rate control.

1) CSMA-CA allocates fair and efficient rates with optimal rate control.
2) The presence of asymmetric edges in a topology leads to a throughput loss larger than the loss caused by any other two-edge topology.
3) The presence of flows that interfere with a common flow but do not interfere with each other deteriorates CSMA-CA's performance.
4) For a chain topology with $n \geq 6$, changing the number of hops does not change CSMA-CA's throughput performance.
5) The presence of multiple congested neighborhoods in a topology degrades CSMA-CA's performance.
6) Among all the multihop topologies we study, the parking lot topology has the worst performance for CSMA-CA.

Note that we have not yet given any explanation into why these interference characteristics improve/degrade CSMA-CA's performance. Recall that the objective of this section was to study the end-to-end performance of CSMA-CA in multihop topologies and determine the interference characteristics that affect CSMA-CA's performance. In Section III, we analytically characterize the neighborhood topology that yields the worst throughput ratio for CSMA-CA over optimal scheduling. This characterization will allow us to provide an explanation for each observation we make in this section.

III. Study of Worst-Case Neighborhood Topologies

In this section, we study the performance of CSMA-CA against the optimal for neighborhood topologies. Specifically, we characterize the neighborhood topology that minimizes the ratio of the throughput achieved by CSMA-CA over that achieved by the optimal at the congested edge at the max-min rate allocation.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

The purpose is twofold. First, we derive the maximum throughput loss that one will observe with CSMA-CA against the optimal in these topologies. Second, this study characterizes the interference characteristics that will degrade CSMA-CA's performance in multihop networks. This characterization will allow us to explain the observations we made in Section II regarding the interference characteristics that affect CSMA-CA's throughput performance.

A. Neighborhood Topologies

Studying neighborhood topologies instead of all possible multihop topologies makes the problem at hand amenable to formal analysis. More importantly, with CSMA-CA, there is a direct one-to-one connection in the throughput performance of neighborhood topologies and multihop topologies: The throughput performance of any multihop topology is dictated by the throughput performance of its *congested neighborhood* topologies. The rationale is similar to wired networks where the throughput performance of a multihop topology is dictated by its congested links [10], [26], [27]. For wireless networks, the idea of congested links gets replaced by congested neighborhoods.

Each flow may pass through several congested neighborhoods. We define the most congested neighborhood a flow passes through to be the neighborhood that gets congested at the lowest throughput among the congested neighborhoods that flow traverses. Thus, there is a unique most congested neighborhood associated with each flow.

The following lemma formally justifies the study of neighborhood topologies to understand the performance of CSMA-CA in multihop networks.

Lemma 3.1: With the CSMA-CA scheduling scheme, a rate allocation that assigns the maximum allowable equal rate to the flows that share the most congested neighborhood[6] is the max-min rate allocation.

Please refer to our prior work [28] for the proof. The lemma states that at the max-min rate allocation, the throughput achieved by a flow is dictated by the most congested neighborhood[7] it passes through. Thus, how two flows interfere with each other[8] can be directly studied by studying how edges interfere in the congested neighborhood shared by both flows.

To simplify the derivation of the formal results in the rest of this section, we restrict the traffic matrix in neighborhood topologies to have only single-hop flows over each edge. We also assume that the arrival process for each flow has independent and identically distributed (i.i.d.) interarrival times and that arrival processes of different flows are independent.

Note that considering single-hop flows only is a very common practice when studying the throughput performance of random access scheduling schemes in multihop networks; see, for example, [11], [14], [15], [17], [20], [25], [31], [33], and references within. That said, the most congested neighborhood of a real-world multihop topology may not satisfy this constraint, e.g., consider a multihop flow whose path has two edges in the most congested neighborhood. However, the single-hop flow assumption is not only quite practical (in its absence, it is almost impossible to get any formal throughput results), but also yields quite accurate results because the main factor affecting the throughput in multihop topologies is the complex interdependence due to the interference between neighboring edges and not the interdependence due to the same flow traversing multiple edges.

B. Preliminaries

1) Model and Notations: This section introduces our notation and assumptions.

As already mentioned, a neighborhood topology is one where there is a particular edge of interest, and all the other edges in the topology interfere with this edge. The edge of interest is assumed to be the congested edge in the neighborhood topology, and we denote it by e_c. The set of neighboring edges is denoted by N_{e_c}. For convenience, we adopt the convention that $e_c \notin N_{e_c}$.

When studying CSMA-CA, we assume that each node uses RTS/CTS because its use is suggested by the IEEE 802.11 standard as it achieves a better scheduling efficiency. Let W_0 and m denote the initial backoff window and the number of exponential backoff windows, respectively. Let T_s denote the time it takes to complete one packet transmission including the time it takes to exchange RTS, CTS, and ACK packets and the headers. We assume that the packet size and the data transmission rate is fixed, hence T_s is a given constant. Again, we assume that the control overhead of optimal scheduling is the same as that of IEEE 802.11.

All the results presented in this paper use the analytical model introduced in our prior work [15]. This model incorporates all protocol components and topological effects like topology asymmetries, DATA and RTS collisions, exponential backoff, virtual and physical carrier sensing, channel losses due to fading, interdependence between both neighboring and nonneighboring edges, etc. Due to space constraints, we do not present the model here. We merely restate the concepts, notations, and the main assumptions that will be used in this paper while deriving the worst-case topology.

The probability that the channel around the transmitter of an edge is busy is defined to be the busy probability. Transmission on any of the neighboring edges that will cause the channel to be sensed busy by Tx_e contributes to the busy probability at e. Let N_e^{busy} denote these set of edges and p_e^{busy} denote the busy probability. For example, edges forming coordinated stations belong to N_e^{busy}.

Transmission on any of the neighboring edges that cannot be sensed by Tx_e but will cause a collision at Rx_e contribute to the RTS collision probability at e. Let N_e^{coll} denote these set of edges. For example, edges forming asymmetric edges with e not being aware of the channel belong to this set. Note that the RTS collision probability also depends on the backoff window value at Tx_e. Let $p_e^{c,i}$, $0 \leq i \leq m$ denote the RTS collision probability when the backoff window value at Tx_e is W_i. Note that $p_e^{c,i}$, $i \geq 1$, is a deterministic function of $p_e^{c,0}$.[9] Hence, among

[6]Recall that the notion of a congested neighborhood has been formally defined in Section II-A.5.

[7]Two flows may pass through the same congested neighborhood. However, if they have different *most congested neighborhoods*, they will have different throughputs at the max-min allocation.

[8]Two flows are said to interfere with each other if they flow through edges that interfere with each other.

[9]Due to space limitations and to avoid significant repetition from our prior work [15], we do not include any details about this function, its derivation, and the associated assumptions here and refer the interested reader to our prior work.

the equations governing the collision probabilities, we need to focus only on those that govern the value of $p_e^{c,0}$. Also, we adopt the convention that whenever we refer to the RTS collision probability from now on, we imply $p_e^{c,0}$.

Finally, we now summarize the important assumptions made in [15].

1) The random process governing the evolution of states at each edge is decoupled to avoid an exploding state space. The dependence between edges is modeled by determining the average values of p_e^{busy} and $p_e^{c,0}$.
2) The impact of the interaction of edges in N_e^{busy} and N_e^{coll} is ignored. Note that a comparison between the analytical and simulation results for several topologies in [15] shows that these two assumptions have a negligible impact on the accuracy of the analytical model in determining the achievable rate region.
3) We now summarize the interference model assumed. Unless explicitly stated, we do not assume a particular physical-layer model to define which nodes can transmit to/interfere with other nodes. However, like prior works [7], [8], [14]–[16], [21], [25], [29], to simplify analysis, interference between nodes is always assumed to be binary—that is, a transmission emanating from one of the interfering nodes will always cause a collision at the other node—and pairwise—that is, interference happens between these node pairs only. Removing these two assumptions complicates analysis to such an extent that even deriving the throughput achieved with optimal scheduling becomes intractable.

2) Topology Characterization: In this section, we define metrics that will be used to characterize the worst-case neighborhood. In the following definitions, a time-slot is equal to the transmission time of one packet.

Definition 3.1 (Scheduling Factor): The number of times the most congested edge e_c is scheduled by optimal scheduling at the max-min allocation is defined to be the scheduling factor of the neighborhood topology.

We denote the scheduling factor of the set of edges in N_{e_c} by $M_{N_{e_c}}$.

Definition 3.2 (Interference Factor): The interference factor is defined as the minimum number of time-slots optimal scheduling will require to schedule all edges in N_{e_c} $M_{N_{e_c}}$ times.

We denote the interference factor of the set of edges in N_{e_c} by $k_{N_{e_c}}$.

The following result follows from the definition of the interference factor.

Lemma 3.2: Optimal scheduling yields a throughput of $M_{N_{e_c}}/(k_{N_{e_c}} + M_{N_{e_c}})T_s$ at edge e_c at the max-min rate allocation.

Note that we will use the performance of optimal scheduling as a baseline and derive the penalty from using CSMA-CA scheduling instead of the optimal. The performance of optimal, and thus of CSMA-CA as well, will be expressed in terms of the scheduling and interference factors. Computing these factors, of course, requires the computation of the max-min schedule. Our goal is to derive the penalty, rather than derive a method to determine these factors, which we accomplish in Section III-C.

C. Worst-Case Neighborhood

In this section, we give an $O(|N_{e_c}|^5)$ algorithm that systematically builds the worst-case neighborhood topology and establishes the worst-case performance degradation from using CSMA-CA instead of optimal scheduling. An important and implicit contribution of our methodology is to identify from the underlying complex analytical model which variables and which portions of the model are affected by the topology characteristics. This allows us to reduce complexity without sacrificing accuracy as the variables that do not depend on the topological characteristics, as well as the equations governing them can be left out from the analysis.

The performance of a given neighborhood topology depends on the following two interference characteristics: 1) $\forall e_i \in N_{e_c}$, how e_i and e_c interfere with each other—in other words, whether e_i and e_c interfere as coordinated stations, or as near hidden edges or asymmetrically or as far hidden edges; 2) $\forall e_i$, $e_j \in N_{e_c}$, $i \neq j$, whether e_i and e_j interfere with each other or not. Note that whether e_i and e_j interfere as coordinated stations or asymmetrically, etc., does not impact the throughput on e_c for either CSMA-CA [15] or optimal scheduling [13].

1) Algorithm: Before presenting the algorithm, we first define four new variables that will be used to describe the algorithm. Let $M_{N_{e_c}^{\text{busy}}}$, $k_{N_{e_c}^{\text{busy}}}$, and $M_{N_{e_c}^{\text{coll}}}$, $k_{N_{e_c}^{\text{coll}}}$ denote the scheduling factor and interference factor for the edges in $N_{e_c}^{\text{busy}}$ and $N_{e_c}^{\text{coll}}$, respectively, given e_c is the congested edge. If none of the edges in $N_{e_c}^{\text{busy}}$ ($N_{e_c}^{\text{coll}}$) interfere with each other, then $k_{N_{e_c}^{\text{busy}}} = M_{N_{e_c}^{\text{busy}}} = 1$ ($k_{N_{e_c}^{\text{coll}}} = M_{N_{e_c}^{\text{coll}}} = 1$); if all the edges in $N_{e_c}^{\text{busy}}$ ($N_{e_c}^{\text{coll}}$) interfere with each other, then $M_{N_{e_c}^{\text{busy}}} = 1$, $k_{N_{e_c}^{\text{busy}}} = |N_{e_c}^{\text{busy}}|$ ($M_{N_{e_c}^{\text{coll}}} = 1$, $k_{N_{e_c}^{\text{coll}}} = |N_{e_c}^{\text{coll}}|$).

The algorithm has two parts. The first part exhaustively searches over the entire space of the number of edges in $N_{e_c}^{\text{coll}}$ (which varies from 0 to $|N_{e_c}|$), $k_{N_{e_c}^{\text{busy}}}$ (which varies from 1 to $|N_{e_c}^{\text{busy}}|$), $k_{N_{e_c}^{\text{coll}}}$ (which varies from 1 to $|N_{e_c}^{\text{coll}}|$), $M_{N_{e_c}^{\text{busy}}}$ (which varies from 1 to $k_{N_{e_c}^{\text{busy}}}$), and $M_{N_{e_c}^{\text{coll}}}$ (which varies from 1 to $k_{N_{e_c}^{\text{coll}}}$). More precisely, if TR_{worst} denotes the throughput ratio of the worst-case neighborhood topology and $\text{TR}_{\text{worst}}(|N_{e_c}^{\text{coll}}|, k_{N_{e_c}^{\text{busy}}}, k_{N_{e_c}^{\text{coll}}}, M_{N_{e_c}^{\text{busy}}}, M_{N_{e_c}^{\text{coll}}})$ denotes the worst-case throughput ratio given $|N_{e_c}^{\text{coll}}|$, $k_{N_{e_c}^{\text{busy}}}$, $k_{N_{e_c}^{\text{coll}}}$, $M_{N_{e_c}^{\text{busy}}}$, and $M_{N_{e_c}^{\text{coll}}}$, then

$$\text{TR}_{\text{worst}} = \min_{|N_{e_c}^{\text{coll}}|} \min_{k_{N_{e_c}^{\text{busy}}}, k_{N_{e_c}^{\text{coll}}}} \min_{M_{N_{e_c}^{\text{busy}}}, M_{e_c N^{\text{coll}}}} \left(\text{TR}_{\text{worst}}\left(|N_{e_c}^{\text{coll}}|, k_{N_{e_c}^{\text{busy}}}, k_{N_{e_c}^{\text{coll}}}, M_{N_{e_c}^{\text{busy}}}, M_{N_{e_c}^{\text{coll}}}\right)\right). \quad (1)$$

The second part of the algorithm determines the value of $\text{TR}_{\text{worst}}(|N_{e_c}^{\text{coll}}|, k_{N_{e_c}^{\text{busy}}}, k_{N_{e_c}^{\text{coll}}}, M_{N_{e_c}^{\text{busy}}}, M_{N_{e_c}^{\text{coll}}})$.

Determining $\text{TR}_{\text{worst}}(|N_{e_c}^{\text{coll}}|, k_{N_{e_c}^{\text{busy}}}, k_{N_{e_c}^{\text{coll}}}, M_{N_{e_c}^{\text{busy}}}, M_{N_{e_c}^{\text{coll}}})$: To determine $\text{TR}_{\text{worst}}(|N_{e_c}^{\text{coll}}|, k_{N_{e_c}^{\text{busy}}}, k_{N_{e_c}^{\text{coll}}}, M_{N_{e_c}^{\text{busy}}}, M_{N_{e_c}^{\text{coll}}})$, we have to derive the two interference characteristics stated at the start of the section. At a high level, for a given scheduling and interference factor, the optimal performance is known. Among all topologies with the same scheduling and interference factors, the worst-case one minimizes the throughput of CSMA-CA or, equivalently, maximizes the busy and collision probabilities. Theorems 3.3 and 3.4 offer a tight upper bound on these probabilities and constitute the crux of the derivation.

We first state a theorem to specify how e_c interferes with all the edges in N_{e_c}.

Theorem 3.1: The worst-case neighborhood topology has the following properties.

a) All edges in $N_{e_c}^{\text{busy}}$ interfere as near hidden edges with e_c.

b) All edges in $N_{e_c}^{\text{coll}}$ interfere as asymmetric edges with e_c.

By definition of $N_{e_c}^{\text{coll}}$, between $e_i \in N_{e_c}^{\text{coll}}$ and e_c, e_c is the edge that is not aware of the channel state.

Since, the number of edges in $N_{e_c}^{\text{coll}}$ and $N_{e_c}^{\text{busy}}$ are known ($|N_{e_c}^{\text{busy}}| = |N_{e_c}| - |N_{e_c}^{\text{coll}}|$), Theorem 3.1 completely specifies how e_c interferes with edges in N_{e_c}. Theorem 3.1 follows directly from the following observations. Among the kind of two-edge topologies belonging to $N_{e_c}^{\text{busy}}$, all of them yield the same busy probability and their respective collision probabilities are independently combined [15]. Thus, near hidden edges that have the largest individual collision probability will cause the maximum throughput loss at e_c. Among the kind of two-edge topologies belonging to $N_{e_c}^{\text{coll}}$, all of them yield the same RTS collision probability, however only the asymmetric topology will also have DATA collisions [15]. Hence, the asymmetric topology will cause the maximum throughput loss at e_c.

We next determine which of the $e_i, e_j \in N_{e_c}$, $i \neq j$, interfere with each other in the worst-case neighborhood topology. We break this task into the following three steps. 1) We first determine how many edge pairs with one edge lying in $N_{e_c}^{\text{busy}}$ and the other lying in $N_{e_c}^{\text{coll}}$ interfere with each other. 2) We next determine which of the $e_i, e_j \in N_{e_c}^{\text{busy}}$, $i \neq j$ interfere with each other. 3) Finally, we determine which of the $e_i, e_j \in N_{e_c}^{\text{coll}}$, $i \neq j$ interfere with each other. We first state a theorem to determine how many edge pairs with one edge lying in $N_{e_c}^{\text{busy}}$ and the other lying in $N_{e_c}^{\text{coll}}$ interfere with each other.

Theorem 3.2: The worst-case neighborhood topology has the following properties.

a) No edge in $N_{e_c}^{\text{busy}}$ interferes with any edge in $N_{e_c}^{\text{coll}}$.

b) The throughput achieved by optimal scheduling is equal to $(1/T_s) \cdot \max(M_{N_{e_c}^{\text{busy}}}/(k_{N_{e_c}^{\text{busy}}} + M_{N_{e_c}^{\text{busy}}}))$, $(M_{N_{e_c}^{\text{coll}}}/(k_{N_{e_c}^{\text{coll}}} + M_{N_{e_c}^{\text{coll}}}))$.

The proof of Theorem 3.2 follows directly from the following observation: If an edge in $N_{e_c}^{\text{busy}}$ interferes with any edge in $N_{e_c}^{\text{coll}}$, there is no change in the throughput at e_c of CSMA-CA (as stated in Section III-B.1, the interaction between edges in $N_{e_c}^{\text{busy}}$ and $N_{e_c}^{\text{coll}}$ has negligible impact on throughput), however the throughput at e_c of optimal scheduling may reduce.

We next state a theorem that determines how many edge pairs in $N_{e_c}^{\text{busy}}$ interfere with each other. We use the following notation in the theorem. Let the set containing the $k_{N_{e_c}^{\text{busy}}}$ maximal independent sets covering all the edges in $N_{e_c}^{\text{busy}} M_{N_{e_c}^{\text{busy}}}$ times be denoted by $\mathcal{S}_I^{N_{e_c}^{\text{busy}}}$. By definition, all the edges belonging to a particular maximal independent set $I_j \in \mathcal{S}_I^{N_{e_c}^{\text{busy}}}$ do not interfere with each other.

Theorem 3.3: The worst-case neighborhood topology has the following properties.

a) Two edges $e_i, e_j \in N_{e_c}^{\text{busy}}$ do not interfere with each other only if there exists a maximal independent set $I_l \in \mathcal{S}_I^{N_{e_c}^{\text{busy}}} : e_i, e_j \in I_l$.

b) $p_{e_c}^{\text{busy}} \leq (k_{N_{e_c}^{\text{busy}}}/M_{N_{e_c}^{\text{busy}}})(1 - (1 - \lambda_{e_c} T_s)^{|N_{e_c}^{\text{busy}}|M_{N_{e_c}^{\text{busy}}}/k_{N_{e_c}^{\text{busy}}}})$, where λ_{e_c} is the packet arrival rate at edge e_c.

The proof of Theorem 3.3 is contained in the Appendix. The bound on $p_{e_c}^{\text{busy}}$ in Theorem 3.3 is exact when $M_{N_{e_c}^{\text{busy}}} = 1$ and $|N_{e_c}^{\text{busy}}|M_{N_{e_c}^{\text{busy}}}/k_{N_{e_c}^{\text{busy}}}$ is an integer. Otherwise, Theorem 3.3(b) can be used to derive a lower bound on the value of $\text{TR}_{\text{worst}}(|N_{e_c}^{\text{coll}}|, k_{N_{e_c}^{\text{busy}}}, k_{N_{e_c}^{\text{coll}}, M_{N_{e_c}^{\text{busy}}}, M_{N_{e_c}}^{\text{coll}}})$.

Finally, we now state a theorem to determine how many edge pairs in $N_{e_c}^{\text{coll}}$ interfere with each other. Let the set containing the $k_{N_{e_c}^{\text{coll}}}$ maximal independent sets covering all the edges in $N_{e_c}^{\text{coll}} M_{N_{e_c}^{\text{coll}}}$ times be denoted by $\mathcal{S}_I^{N_{e_c}^{\text{coll}}}$.

Theorem 3.4: The worst-case neighborhood topology has the following properties.

a) Two edges $e_i, e_j \in N_{e_c}^{\text{coll}}$ do not interfere with each other only if there exists a maximal independent set $I_l \in \mathcal{S}_I^{N_{e_c}^{\text{coll}}}$: $e_i, e_j \in I_l$.

b) $p_{e_c}^{c,0} \leq (k_{N_{e_c}^{\text{coll}}}/M_{N_{e_c}^{\text{busy}}})(1 - (1 - \lambda_{e_c} T_s)^{|N_{e_c}^{\text{coll}}|M_{N_{e_c}^{\text{coll}}}/k_{N_{e_c}^{\text{coll}}}})$, where λ_{e_c} is the packet arrival rate at edge e_c.

Recall that $p_{e_c}^{c,0}$ denotes the RTS collision probability. The proof of Theorem 3.4 follows along similar lines as the proof of Theorem 3.3, and is skipped for brevity. (Note that the expressions governing $p_{e_c}^{\text{busy}}$ and $p_{e_c}^{c,0}$ are similar, hence a similar derivation yields both Theorems 3.3 and 3.4.) Also, similar to Theorem 3.3, the bound on $p_{e_c}^{c,0}$ is exact only when $M_{N_{e_c}^{\text{coll}}} = 1$ and $|N_{e_c}^{\text{coll}}|M_{N_{e_c}^{\text{coll}}}/k_{N_{e_c}^{\text{coll}}}$ is an integer, otherwise Theorem 3.4(b) can be used to derive a lower bound on $\text{TR}_{\text{worst}}(|N_{e_c}^{\text{coll}}|, k_{N_{e_c}^{\text{busy}}}, k_{N_{e_c}^{\text{coll}}}, M_{N_{e_c}^{\text{busy}}}, M_{N_{e_c}^{\text{coll}}})$.

Using the upper bounds on $p_{e_c}^{\text{busy}}$ and $p_{e_c}^{c,0}$ from Theorems 3.3 and 3.4, we derive a lower bound on $\text{TR}_{\text{worst}}(|N_{e_c}^{\text{coll}}|, k_{N_{e_c}^{\text{busy}}}, k_{N_{e_c}^{\text{coll}}}, M_{N_{e_c}^{\text{busy}}}, M_{N_{e_c}^{\text{coll}}})$. Applying these bounds to evaluate (1), we find that the lower bound on TR_{worst} always occurred when $M_{N_{e_c}^{\text{busy}}} = M_{N_{e_c}^{\text{coll}}} = 1$ and $|N_{e_c}^{\text{busy}}|M_{N_{e_c}^{\text{busy}}}/k_{N_{e_c}^{\text{busy}}}$, and $|N_{e_c}^{\text{coll}}|M_{N_{e_c}^{\text{coll}}}/k_{N_{e_c}^{\text{coll}}}$ are integers. Recall that the bound is exact for these conditions.

As a final note, to simplify the derivation of the bounds for the idle and collision probabilities at the congested edge [see Theorems 3.3(b) and 3.4(b) and their proof in Appendix A], we are assuming there is only one flow per edge. Appendix B extends the algorithm to the case of multiple flows per edge. It does so by presenting a method to divide edges with multiple flows into multiple edges with a single flow each and by formally establishing that the resulting topology yields the same throughput ratio at the congested edge as the original topology (with multiple flows per edge).

2) Numerical Results: The solid line with dots in Fig. 4 plots the worst-case throughput ratio as a function of the number of neighboring edges ($|N_{e_c}|$) for the default parameters of IEEE 802.11(b) in Qualnet [2] and a data rate of 1 Mb/s. (Unless explicitly stated, all the numerical results presented in this section assume these system parameters.) Note that in typical topologies, one expects that the number of neighboring edges per neighborhood will be less than 20. Hence, we display the plot until $N_{e_c} = 20$. The ratio keeps on decreasing as $|N_{e_c}|$ increases, and it is slightly larger than 16% for $|N_{e_c}| = 20$. Fig. 4 also plots the worst-case throughput ratio of maximal scheduling derived in [8] (solid line with rhombus), and we observe that CSMA-CA significantly outperforms maximal

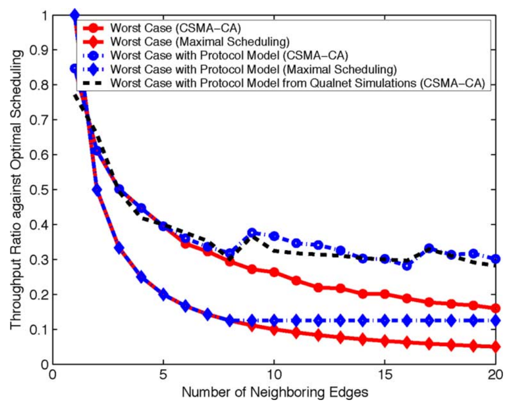

Fig. 4. Worst-case and average throughput ratios for CSMA-CA and maximal scheduling against optimal scheduling for different neighborhood sizes.

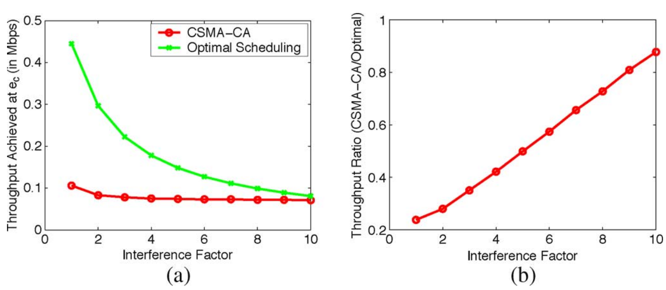

Fig. 5. (a) Worst-case throughput achieved at e_c as a function of the interference factor for $|N_{e_c}| = 10, M_{N_{e_c}} = 1$. (b) Worst-case throughput ratio achieved at e_c as a function of the interference factor for $|N_{e_c}| = 10, M_{N_{e_c}} = 1$.

scheduling. (The other plots in this figure will be derived in Section III-D.)

Now, we present the value of $|N_{e_c}^{\rm coll}|$, the scheduling factor and the interference factor as determined by the first part of the algorithm for the worst-case topology. $|N_{e_c}^{\rm coll}|$ for the worst-case topology is always equal to $|N_{e_c}|$. This is not surprising as collisions cause exponential backoffs reducing the throughput drastically. The scheduling factor and the interference factor for the worst-case topology are always equal to 1 irrespective of the value of $|N_{e_c}|$.[10] This is surprising, as a smaller interference factor implies fewer edges interfering with each other, which should actually improve the throughput performance of CSMA-CA. However, note that optimal scheduling also has a better throughput performance for smaller interference factors. Thus, decreasing the interference factor improves the performance of both scheduling schemes. However, the improvement is more significant for optimal scheduling than CSMA-CA because CSMA-CA schedules noninterfering neighboring edges independently rather than simultaneously. Fig. 5(a) and (b) shows this by plotting the worst-case throughput achieved at e_c for CSMA-CA and optimal scheduling and the corresponding ratio for different values of the interference factor for $M_{N_{e_c}} = 1$.

[10]We numerically verify that these two characteristics hold until $|N_{e_c}| \leq 1000$. Also, topologies with the scheduling factor >1 but equal to the interference factor yield the same bound. However, for these topologies, the ratio computed is a lower bound, while for the topology with $\text{scheduling factor} = 1$, this is the exact achieved value.

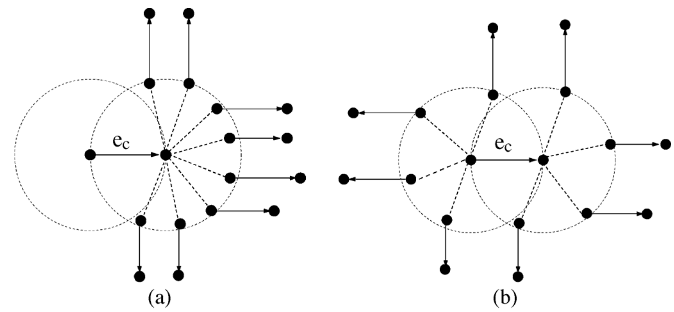

Fig. 6. Worst-case topology for $|N_{e_c}| = 8$. (a) Worst case with no assumptions on the physical layer. (b) Worst case assuming the protocol model of interference.

As an example of how the worst-case topology looks like, we plot this topology for $|N_{e_c}| = 8$ in Fig. 6(a). It contains eight noninterfering asymmetric edges.

D. Imposing Practical Constraints

The worst-case topology derived in the previous section might not be constructible in practice due to the geometrical constraints imposed by the physical layer. Hence, the worst-case performance of CSMA-CA for *practical* topologies should be better than the one derived in the previous section. Note that we still studied the worst-case performance of CSMA-CA with no physical-layer constraints because of its following two advantages: 1) it establishes absolute worst-case bounds; and 2) it gives an analytical explanation of the surprising observation that the throughput ratio of CSMA-CA over optimal improves as the interference factor increases.

In addition to the assumptions specified in Section III-B.1, in this section, we also assume a particular physical-layer model. Bounds on deterministic scheduling schemes like maximal scheduling and greedy maximal scheduling were derived assuming the protocol model of interference for analytical tractability [8], [16]. To be able to make a fair comparison as well as maintain analytical tractability, we also use the protocol model of interference. Setting the interference range in the protocol model is important to get realistic results. In this section, instead of setting the interference range so as to get more realistic results, we set the interference range to get the worst-case bounds on CSMA-CA. In other words, we choose the interference range so that the worst-case throughput ratio is minimized. Thus, with more realistic interference range settings, the performance of CSMA-CA would be even better.

In the previous section, we proved that the more the interference is in the topology, the better CSMA-CA's throughput ratio is. Thus, we should set the interference range to its minimum possible value to minimize interference. Hence, we set the interference range to be equal to the transmission range.

Our assumptions on the physical layer impose two constraints on how edges in N_{e_c} can interfere with each other.

1) Noninterfering Neighbors: The first constraint bounds the maximum number of noninterfering neighboring edges of e_c. Using a derivation similar to [8], we derive the following two rules. 1) The maximum number of nodes that interfere with Rx_{e_c} but not Tx_{e_c}, as well as do not interfere with each other is equal to 4. This rule bounds the maximum number of noninterfering edges in $N_{e_c}^{\rm coll}$ to be 4. Thus, the worst-case topology

TABLE III
DEFINING THE SECOND CONSTRAINT

Cardinality of of C_1	Cardinality of C_2	Number of edges in C_2 which an edge in C_1 can interfere with
$\|C_1\| = 1$	$\|C_2\| \geq 1$	$\|C_2\|$
$\|C_1\| > 1$	$\|C_2\| = 1$	1
$\|C_1\| > 1$	$\|C_2\| > 1$	2

derived in the previous section is not feasible for $|N_{e_c}| > 4$. 2) The maximum number of nodes that interfere with either Tx_{e_c} or Rx_{e_c} but do not interfere with each other is equal to 8. This rule implies that any neighborhood with more than eight edges cannot have an interference factor equal to 1.

2) Interference Between Edges Belonging to Different Maximal Independent Sets: Let $I_1, I_2 \in \mathcal{S}_I^{N_{e_c}}$. Let $C_j, j = 1, 2$ denote the set of edges in I_j that also belong to $N_{e_c}^{\text{coll}}$. The number of edge pairs with one edge in C_1 and the other edge in C_2 that interfere with each other effects the throughput ratio. Theorem 3.4(a) proves that all such edge pairs interfere with each other in the worst-case neighborhood topology derived without incorporating physical-layer constraints.

The second constraint bounds the maximum number of interfering edge pairs with edges belonging to different maximal independent sets. Table III states the maximum number of edges in C_2 that an edge in C_1 can interfere with for different cardinalities of C_1 and C_2. These values are derived by writing the geometrical constraints imposed on the transmitter and receiver of these edges and checking if a solution exists.

Now, let $B_j, j = 1, 2$ denote the set of edges in I_j that also belong to $N_{e_c}^{\text{busy}}$. In a similar fashion, we derive that the maximum number of edges in B_2 that an edge in B_1 can interfere with is governed by Table III also.

3) Numerical Results: We now look at the worst-case throughput ratio with the protocol model of interference. We construct the worst-case topology using brute force by constructing all possible topologies allowed after imposing these two constraints, evaluating the performance of CSMA-CA and optimal scheduling for each of these topologies and finding the one that has the worst throughput ratio. Thus, results for this section are derived by numerically solving the analytical model after imposing the two constraints derived in Sections III-D.1 and III-D.2. This procedure formally establishes the worst-case bounds for CSMA-CA with the protocol model. Note that having an analytical model allows us to quickly evaluate the performance of CSMA-CA for each topology. If one had to evaluate the performance using ns-2 or Qualnet simulations, this approach will become prohibitively expensive. For example, on a 3.06-GHz Linux box, finding the worst-case neighborhood topology for $|N_{e_c}| = 4$ using Qualnet simulations will take more than 800 h, and this time will exponentially increase as the number of neighboring edges increases.

The dashed line with dots in Fig. 4 plots the worst-case throughput ratio of CSMA-CA for the protocol model. The worst-case performance never goes below 30% with the constraints imposed by the protocol model. We also see jumps at multiples of 8 because the maximum number of noninterfering neighbors (which can be scheduled simultaneously) e_c can have is equal to 8; hence, after a multiple of 8, optimal scheduling takes one extra slot to schedule, which deteriorates its throughput performance. Note that the performance of CSMA-CA also deteriorates, but not as much as optimal scheduling, hence the observed jump at multiples of 8. Fig. 4 also plots the worst-case throughput ratio of maximal scheduling (dashed line with rhombus), which is equal to 12.5%.

4) Characterizing the Worst-Case Topology: Recall that the worst-case neighborhood topology derived in Section III without imposing any geometrical constraints from the physical layer had the following four characteristics: 1) lowest interference factor possible; 2) scheduling factor is equal to 1; 3) uniformly distributing edges among the maximal independent sets; 4) all neighboring edges belonged to $N_{e_c}^{\text{coll}}$. Now we characterize the worst-case topology derived with the physical-layer model assumed in this section. The intuition derived in Section III for the first three characteristics still holds. We numerically verify that the worst-case topology has these three characteristics for all values of $|N_{e_c}| \leq 20$.

The last characteristic gets modified slightly. We explain the reason using an example. Let us consider the worst-case neighborhood topology for $|N_{e_c}| = 5$. Recall that the maximum number of noninterfering edges possible in $N_{e_c}^{\text{coll}}$ is 4, but the minimum possible interference factor for $|N_{e_c}| = 5$ is equal to 1 (see Section III-D.1). Thus, five edges cannot be placed in $N_{e_c}^{\text{coll}}$ while maintaining the interference factor to be 1. Hence, the third characteristic now becomes the following: For a given N_{e_c} and interference factor, $N_{e_c}^{\text{coll}}$ contains as many edges as possible, and the remaining edges are contained in $N_{e_c}^{\text{busy}}$. Thus, for $|N_{e_c}| = 5$, $N_{e_c}^{\text{coll}}$ contains four edges, and $N_{e_c}^{\text{busy}}$ contains the fifth one. We numerically verified this characteristic for all values of $|N_{e_c}| \leq 20$.

As an example of how the worst-case topology looks like, we plot this topology for $|N_{e_c}| = 8$ in Fig. 6(b). There are four asymmetric edges, four coordinated station edges, and none of the edges in N_{e_c} interfere with each other.

E. Summary

To summarize, the presence of the following two interference characteristics in the congested neighborhood leads to a worse performance with CSMA-CA: 1) presence of noninterfering edges; 2) presence of asymmetric edges. These two characteristics explain every observation we made in Section II on the performance of CSMA-CA in multihop topologies.

1) In the fork topology [Fig. 2(g)] and chain-cross [Fig. 3(j)], we observed that the presence of flows in congested neighborhoods that interfere with a common flow but do not interfere with each other deteriorates CSMA-CA's performance. The reason directly follows from the first characteristic that noninterfering edges in the congested neighborhood degrades CSMA-CA's performance.
2) In the chain topology, the interference characteristics in the congested neighborhood are different for each value of n, $n < 6$. It remains the same for all n, $n \geq 6$. Hence, the throughput ratio does not change for the chain topology with $6 \leq n \leq 15$.
3) The chain with two interfering short flows [Fig. 3(d)] achieves better throughputs than the chain topology [Fig. 3(a)]. The reason being that the congested edge changes to $1 \rightarrow 2$ from edge $3 \rightarrow 4$. $N_{3\rightarrow 4}$ has fewer

edges but more noninterfering edges, while $N_{1\to2}$ has more edges, but all of them interfere with each other. Thus, the first characteristic implies that CSMA-CA will have a better performance for the chain with two interfering short flows topology.

4) The chain with three interfering short flows [Fig. 3(g)] achieves worse throughputs than the chain with two interfering short flows [Fig. 3(d)]. Also, the actual throughput values for flows $1 \to 7$, $2 \to 8$, and $2 \to 9$ are the same in both the topologies. Thus, the lower throughput achieved for flow $7 \to 10$ is the cause of the throughput loss. The reason being that the second congested edge is $5 \to 8$ with which edge $7 \to 10$ interferes *asymmetrically*.

F. Discussion

In this section, we briefly discuss some additional results to support our thesis.

Typical Topologies: To understand how CSMA-CA does for typical topologies instead of worst-case ones, we use simulations to derive the average throughput ratio of CSMA-CA and optimal scheduling by simulating 2000 randomly generated neighborhood topologies. We found the average throughput ratio to be always within 55% for topologies with less than 20 neighbors. We also study the throughput ratio achieved in a real topology of an outdoor residential deployment in a Houston, TX, neighborhood [6]. At the max-min allocation, the throughput ratio was found to be equal to 66.8%, which, not surprisingly, is closer to the average ratio rather than the worst-case ratio.

Low Overhead of CSMA-CA: CSMA-CA exchanges only two control messages, namely RTS and CTS, per packet transmission. For most multihop topologies, by adding more control overhead, theoretically it is possible to improve CSMA-CA's throughput performance. For example, if one can implement greedy maximal scheduling with zero overhead, one can get close to 100% of the optimal for both the stack topology and the chain topology introduced in Section II-B. This fact is observed through simulations similar to ones performed in [17]. To compare the two scheduling schemes after incorporating the control overhead, we implement the distributed algorithm proposed to implement greedy maximal scheduling [18]. At the max-min allocation, greedy maximal scheduling and CSMA-CA allocate 0.134 and 0.14 Mb/s per flow, respectively, for the flow in the middle topology and allocate 0.047 and 0.06 Mb/s per flow, respectively, for the chain topology. Thus, CSMA-CA outperforms a scheme as efficient as greedy maximal scheduling in real multihop topologies as it achieves good throughputs with low overhead.

More Realistic Interference Range Setting: To understand how much throughput ratio we gain by assuming a more realistic interference range setting in the protocol interference model, we derive the worst-case throughput ratio when the interference range is twice the transmission range, which is the most popular assumption on the value of the interference range [35]. We find the throughput ratio to be always within 38%, which, as expected, is better than the throughput ratio of 30% achieved when the interference range is equal to the transmission range.

Accuracy of the Model Used: The accuracy of our results directly depends on the accuracy of the model proposed in [15]. In addition to the model verification presented in [15], we also verify the accuracy by deriving the throughput of CSMA-CA at the congested edge (e_c) at the max-min allocation from Qualnet simulations in the worst-case neighborhood topology found in Section III-D. The corresponding throughput ratio is plotted in Fig. 4. Note that we do not verify the curve for the worst case with no physical-layer assumptions as the topology cannot be constructed in practice. The small variations in throughput ratio is due to the approximations made in the analytical model of [15] to simplify analysis.

Complexity of Rate Control: One may wonder whether the high performance of a low-overhead scheduler like CSMA-CA is achieved at the expense of a high-overhead rate controller. Interestingly, close-to-optimal throughputs over CSMA-CA in multihop networks are achievable with low-complexity distributed rate control algorithms similar to TCP; see, for example, [28] and [34].

IV. Conclusion

This paper establishes that CSMA-CA achieves fair and efficient throughputs in multihop networks by characterizing the worst-case throughput bounds for CSMA-CA in one-hop neighborhood topologies. We observe that CSMA-CA easily outperforms maximal scheduling and achieves worst-case performance close to greedy maximal scheduling, which is one of the best known approximately optimal scheduling algorithms. The results presented in this paper motivate the use of random access schedulers in single- and multihop wireless networks and prompt researchers to investigate the design of practical congestion control and rate allocation protocols that can realize this good performance over random access schemes.

Appendix A

Whether edges in $N_{e_c}^{\text{busy}}$ interfere with each other or not affects only the value of $p_{e_c}^{\text{busy}}$ (the busy probability at e_c) [15].

Minimizing the throughput achieved by IEEE 802.11 at e_c implies maximizing $p_{e_c}^{\text{busy}}$. We first prove subcase (a). If two edges in $N_{e_c}^{\text{busy}}$ interfere with each other, then the busy probability at e_c is higher than when they do not interfere with each other. By definition, the edges belonging to a $I \in \mathcal{S}_I^{N_{e_c}^{\text{busy}}}$ do not interfere with each other. Since there is no such restriction on distinct edges belonging to different maximal independent sets, letting them interfere with each other increases the busy probability.

We next prove that the throughput ratio for the worst-case topology with scheduling factor equal to 1 and the interference factor equal to $k_{N_{e_c}^{\text{busy}}}/M_{N_{e_c}^{\text{busy}}}$ is smaller than the throughput ratio for the worst-case topology with scheduling factor equal to $M_{N_{e_c}^{\text{busy}}}$ and interference factor equal to $k_{N_{e_c}^{\text{busy}}}$. The throughput of optimal scheduling is the same for both the topologies. For the latter topology, all edges $e \in N_{e_c}^{\text{busy}}$ have to be a part of at least $M_{N_{e_c}^{\text{busy}}}$ maximal independent sets. Now, the more the number of edges e interferes with, the larger is the busy probability, and smaller is the throughput of CSMA-CA. Solving the optimization problem to maximize the number of edge pairs that interfere with each other in the latter topology yields that there are $k_{N_{e_c}^{\text{busy}}}/M_{N_{e_c}^{\text{busy}}}$ maximal independent sets of size

$|N_{e_c}^{\text{busy}}|M_{N_{e_c}^{\text{busy}}}/k_{N_{e_c}^{\text{busy}}}$, which turns out to be the same as the former topology as shown at the end of the proof.

Next, to prove subcase (b), we derive a lower bound on the busy probability of a topology with scheduling factor equal to 1 and the interference factor equal to $k_{N_{e_c}^{\text{busy}}}/M_{N_{e_c}^{\text{busy}}}$.

Lemma A.1: Increasing the rate at any of the edges in $N_{e_c}^{\text{busy}}$ reduces the rate at e_c.

This lemma directly follows from the observation that the busy probability at e_c is a monotonically increasing function of the rate at $e_i, \forall e_i \in N_{e_c}^{\text{busy}}$ [15].

Lemma A.2: For a rate allocation assigning the maximum possible equal rate to all edges in $N_{e_c}^{\text{busy}} \cup e_c$, there always exists a neighborhood topology with the worst-case throughput ratio at e_c with e_c being the congested edge.

This lemma follows directly from the definition of congested neighborhood.

Lemma A.3: IEEE 802.11 allocates equal rates to all the edges in the worst-case neighborhood topology at the max-min allocation.

Proof: Given the rate allocation that assigns the maximum possible equal rate to all the edges in $N_{e_c}^{\text{busy}} \cup e_c$, by Lemmas A.1 and A.2, the rate at none of the edges can be increased without either reducing the rate at e_c or making the system unstable. Thus, it is a max-min rate allocation. ■

At the max-min allocation, all edges in $N_{e_c}^{\text{busy}}$ have a rate equal to λ_{e_c}. Using the results from [15], we derive $p_{e_c}^{\text{busy}} = \sum_{I_j \in \mathcal{S}_I^{N_{e_c}^{\text{busy}}}} 1 - (1 - \lambda_{e_c} T_s)^{|I_j|}$. Maximizing $p_{e_c}^{\text{busy}}$ under the constraint $\sum_{I_j \in \mathcal{S}_I^{N_{e_c}^{\text{busy}}}} |I_j| = |N_{e_c}^{\text{busy}}|$ yields $|I_j| = |N_{e_c}^{\text{busy}}|M_{N_{e_c}^{\text{busy}}}/k_{N_{e_c}^{\text{busy}}}, \forall I_j \in \mathcal{S}_I^{N_{e_c}^{\text{busy}}}$. Thus, the maximum value of $p_{e_c}^{\text{busy}}$ is equal to $(k_{N_{e_c}^{\text{busy}}}/M_{N_{e_c}^{\text{busy}}})(1 - (1 - \lambda_{e_c} T_s)^{|N_{e_c}^{\text{busy}}|M_{N_{e_c}^{\text{busy}}}/k_{N_{e_c}^{\text{busy}}}})$.

Appendix B

In this appendix, we discuss how the algorithm described in Section III-C.1 can be used to study neighborhood topologies with multiple flows per edge.

We first discuss the case where an edge in $N_{e_c}^{\text{busy}}$ has $k > 1$ flows passing through it. We construct a new equivalent neighborhood topology by breaking this edge into k new edges, where each of these newly created edges has one flow passing through it, it interferes with every other edge in the remaining neighborhood topology in exactly the same manner as the original edge, and each of these newly created edges' transmitters can talk to every other new edge's transmitter with zero propagation delay. We will prove that the throughput ratio at the congested edge is the same in both topologies.

Each maximal independent set in the original topology that contains the old multiflow edge will have k corresponding sets in the new topology, each with one of the k new edges because: 1) the interference patterns for this new edge with every other edge is the same as before; and 2) these new edges interfere with each other, hence they cannot belong together in any maximal independent set. Each of these maximal independent sets in the original topology can be used by optimal scheduling to schedule none, one, or multiple flows on the old multiflow edge. Let us say one of these sets is being used to schedule one of the flows in the original topology, and this flow passes through the ith new edge, $i \leq k$, in the new topology. Since a corresponding set exists in the new topology having the ith edge, this set can be scheduled to satisfy the flow with no impact on other edges. Thus, the original optimal schedule directly yields the optimal schedule for this new topology with no impact on the throughput of other edges including the congested edge.

Next, we prove that the throughput on the congested edge does not change under CSMA-CA as well. An edge in $N_{e_c}^{\text{busy}}$ impacts the throughput on e_c, the congested edge, only by changing $p_{e_c}^{\text{busy}}$. We prove that the value of $p_{e_c}^{\text{busy}}$ remains the same in both topologies, and hence the throughput on e_c remains the same. Reference [15, Lemma 14] derives the value of idle probability, and hence equivalently the busy probability on a given edge. As per this result, $p_{e_c}^{\text{busy}}$ is governed by $P(\cup_{e_n \in N_{e_c}^{\text{busy}}} X_{e_n})$, which is derived in [15, Lemma 11]. X_{e_n} denotes that there is a transmission going on at edge e_n and is derived in [15, p. 1125] before Lemma 11 to be $K_{e_n,T}\lambda_{e_n}T_s$, where $K_{e_n,T}$ is derived in [15, p. 1122] before Lemma 3 and depends on the DATA collision probability on e_n, λ_{e_n} is the arrival rate on e_n and T_s is the packet transmission time that is a constant for a fixed size packet. Let us consider $\mathcal{N}_{\text{subset}}^i$, which is a set containing all subsets of $N_{e_c}^{\text{busy}}$ having exactly $i \leq |N_{e_c}^{\text{busy}}|$ edges. Then, $\sum_{N_{\text{subset}} \in \mathcal{N}_{\text{subset}}^i} P(\cap_{e_i \in N_{\text{subset}}} X_{e_i})$ remains the same in both the topologies for all values of i because: 1) the new edges have exactly the same DATA collision probabilities, and hence the same $K_{e_n,T}$ as the original edge (since RTS-CTS is used, no extra DATA collisions will occur due to the addition of these extra edges); and 2) the sum of the arrival rates on these k new edges is the same as the arrival rate on the original edge. Hence, $P(\cup_{e_n \in N_{e_c}^{\text{busy}}} X_{e_n})$ that is derived using $\sum_{N_{\text{subset}} \in \mathcal{N}_{\text{subset}}^i} P(\cap_{e_i \in N_{\text{subset}}} X_{e_i})$ remains the same, which implies $p_{e_c}^{\text{busy}}$ remains the same in both topologies. This proves that a topology with multiple flows per edge for edges in $N_{e_c}^{\text{busy}}$ can be replaced by an equivalent topology with one flow per edge.

A similar argument holds for splitting an edge in $N_{e_c}^{\text{coll}}$ with multiple flows into edges with a single flow each because the expression governing $p_{e_c}^{c,0}$ as derived in [15, Lemma 13] is very similar to the expression for $p_{e_c}^{\text{busy}}$, the only difference being that it will depend on edges in $N_{e_c}^{\text{coll}}$ and not in $N_{e_c}^{\text{busy}}$.

Finally, there can be multiple flows going through the congested edge. This has no impact on the algorithm described in Section III-C.1, as nothing in the algorithm precludes the congested edge having multiple flows.

To summarize, a neighborhood topology with multiflow edges can be represented by an equivalent neighborhood topology with only single-flow edges in N_{e_c} and having the same throughput ratio, and hence the algorithm of Section III-C.1 can be used to derive the worst-case throughput ratio for neighborhood topologies with multiflow edges as well.

References

[1] *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications—Higher-speed Physical Layer Extension in the 2.4 GHz Band*, IEEE Std 802.11b-1999, Nov. 2002.

[2] Qualnet, Los Angeles, CA, "Network simulator," Accessed Oct. 2007 [Online]. Available: http://www.qualnet.com

[3] D. P. Bertsekas and R. G. Gallagher, *Data Networks*, 2nd ed. Upper Saddle River, NJ: Prentice-Hall, 1992.

[4] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 3, pp. 535–547, Mar. 2000.

[5] M. Brodsky and R. Morris, "In defense of wireless carrier sense," in *Proc. ACM SIGCOMM*, 2009, pp. 147–158.

[6] J. Camp, J. Robinson, C. Steger, and E. Knightly, "Measurement driven deployment of a two-tier urban mesh access network," in *Proc. ACM MobiSys*, 2006, pp. 96–109.

[7] D. Chafekar, D. Levin, V. A. Kumar, M. Marathe, S. Parthasarathy, and A. Srinivasan, "Capacity of asynchronous random-access scheduling in wireless networks," in *Proc. IEEE INFOCOM*, 2008, pp. 1148–1156.

[8] P. Chaporkar, K. Kar, X. Luo, and S. Sarkar, "Throughput and fairness guarantees through maximal scheduling in wireless networks," *IEEE Trans. Inf. Theory*, vol. 54, no. 2, pp. 572–594, Feb. 2008.

[9] D. Clark, "Window and acknowledgement strategy in TCP," Internet Engineering Task Force, RFC 813, 1982.

[10] S. Floyd, "Connections with multiple congested gateways in packet-switched networks part 1: One-way traffic," *Comput. Commun. Rev.*, vol. 21, no. 5, pp. 30–47, 1991.

[11] M. Garetto, T. Salonidis, and E. Knightly, "Modeling per-flow throughput and capturing starvation in CSMA multi-hop wireless networks," in *Proc. IEEE INFOCOM*, 2006, pp. 1–13.

[12] A. Gupta, X. Lin, and R. Srikant, "Low-complexity distributed scheduling algorithms for wireless networks," in *Proc. IEEE INFOCOM*, 2007, pp. 1631–1639.

[13] K. Jain, J. Padhye, V. Padmanabhan, and L. Qiu, "Impact of interference on multi-hop wireless network performance," in *Proc. ACM MobiCom*, 2003, pp. 66–80.

[14] L. Jiang and J. Walrand, "A distributed CSMA algorithm for throughput and utility maximization in wireless networks," in *Proc. 46th Annu. Allerton Conf. Commun., Control, Comput.*, 2008, pp. 1511–1519.

[15] A. Jindal and K. Psounis, "The achievable rate region of 802.11-scheduled multihop networks," *IEEE/ACM Trans. Networking*, vol. 17, no. 4, pp. 1118–1131, 2009.

[16] C. Joo, X. Lin, and N. Shroff, "Understanding the capacity region of the greedy maximal scheduling algorithm in multi-hop wireless networks," in *Proc. IEEE INFOCOM*, 2008, pp. 1103–1111.

[17] C. Joo and N. Shroff, "Performance of random access scheduling schemes in multi-hop wireless network," in *Proc. IEEE INFOCOM*, 2007, pp. 19–27.

[18] M. Leconte, J. Ni, and R. Srikant, "Improved bounds on the throughput efficiency of greedy maximal scheduling in wireless networks," in *Proc. ACM MobiHoc*, 2009, pp. 165–174.

[19] J. Li, C. Blake, D. D. Couto, H. Lee, and R. Morris, "Capacity of ad hoc wireless networks," in *Proc. ACM MobiCom*, 2001, pp. 61–69.

[20] K. Medepalli and F. A. Tobagi, "Towards performance modeling of IEEE 802.11 based wireless networks: A unified framework and its applications," in *Proc. IEEE INFOCOM*, 2006, pp. 1–12.

[21] E. Modiano, D. Shah, and G. Zussman, "Maximizing throughput in wireless networks via gossip," in *Proc. ACM SIGMETRICS*, 2006, pp. 27–38.

[22] R. Motwani and P. Raghavan, *Randomized Algorithms*, 1st ed. Cambridge, U.K.: Cambridge Univ. Press, 1995.

[23] J. Nagle, "Congestion control in IP/TCP Internetworks," Internet Engineering Task Force, RFC 896, 1984.

[24] K. Nahm, A. Helmy, and C. J. Kuo, "TCP over multihop 802.11 networks: Issues and performance enhancements," in *Proc. ACM MobiHoc*, 2005, pp. 277–287.

[25] J. Ni and R. Srikant, "Distributed CSMA/CA algorithms for achieving maximum throughput in wireless networks," in *Proc. Inf. Theory Appl. Workshop*, 2009, p. 250.

[26] F. Papadopoulos, K. Psounis, and R. Govindan, "Performance preserving topological downscaling of internet-like networks," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 12, pp. 2313–2326, Dec. 2006.

[27] K. Papagiannaki, "Provisioning IP backbone networks based on measurements," Ph.D. dissertation, University College London, London, U.K., 2003.

[28] S. Rangwala, A. Jindal, K. Jang, K. Psounis, and R. Govindan, "Neighborhood-centric congestion control for multi-hop wireless mesh networks," *IEEE/ACM Trans. Netw.*, 2009, accepted for publication.

[29] S. Sanghavi, L. Bui, and R. Srikant, "Distributed link scheduling with constant overhead," in *Proc. ACM SIGMETRICS*, 2007, pp. 313–324.

[30] G. Sharma, R. Mazumdar, and N. Shroff, "On the complexity of scheduling in wireless networks," in *Proc. ACM MobiCom*, 2006, pp. 227–238.

[31] J. Shin, D. Shah, and S. Rajagopalan, "Network adiabatic theorem: An efficient randomized protocol for contention resolution," in *Proc. ACM SIGMETRICS*, 2009, pp. 133–144.

[32] K. Tan, F. Jiang, Q. Zhang, and X. Shen, "Congestion control in multihop wireless networks," *IEEE Trans. Veh. Technol.*, vol. 56, no. 2, pp. 863–873, Mar. 2006.

[33] X. Wang and K. Kar, "Throughput modeling and fairness issues in CSMA/CA based Ad-hoc networks," in *Proc. IEEE INFOCOM*, 2005, vol. 1, pp. 23–34.

[34] A. Warrier, S. Janakiraman, S. Ha, and I. Rhee, "DiffQ: Practical differential backlog congestion control for wireless networks," in *Proc. IEEE INFOCOM*, 2009, pp. 262–270.

[35] K. Xu, M. Gerla, and S. Bae, "How effective is the IEEE 802.11 RTS/CTS handshake in ad hoc networks?," in *Proc. IEEE GLOBECOM*, 2002, vol. 1, pp. 72–76.

[36] K. Xu, M. Gerla, L. Qi, and Y. Shu, "TCP unfairness in ad-hoc wireless networks and neighborhood RED solution," in *Proc. ACM MobiCom*, 2003.

Apoorva Jindal (M'08) received the B.Tech. degree from the Indian Institute of Technology, Kanpur, India, in 1998, and the M.S. and Ph.D. degrees from the University of Southern California, Los Angeles, in 2007 and 2009, respectively, all in electrical engineering.

He is currently a Member of Technical Staff with Juniper Networks, Inc., San Jose, CA. His research interests include performance analysis and design of protocols for multihop wireless networks.

Konstantinos Psounis (S'97–M'02–SM'08) received the first degree in electrical engineering and computer science from the National Technical University of Athens, Athens, Greece, in 1997, and the M.S. and Ph.D. degrees in electrical engineering from Stanford University, Stanford, CA, in 1999 and 2002, respectively.

He is an Associate Professor of electrical engineering and computer science with the University of Southern California, Los Angeles.