Making the Case for Random Access Scheduling in Wireless Multi-hop Networks

Apoorva Jindal
Electrical Engineering and Computer Science
University of Michigan
Ann Arbor, Michigan, 48109
Email: apoorvaj@usc.edu

Konstantinos Psounis
Department of Electrical Engineering
University of Southern California
Los Angeles, California, 90089
Email: kpsounis@usc.edu

***Abstract*—This paper formally establishes that random access scheduling schemes, and, more specifically CSMA-CA, yields exceptionally good performance in the context of wireless multi-hop 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 which grossly underutilize the available capacity that random access provides. To establish our thesis we compare the max-min rate allocation achieved by CSMA-CA and optimal in multi-hop topologies and find that: (i) CSMA-CA is never worse than 16% of the optimal when ignoring physical layer constraints, (ii) 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.**

I. Introduction

A fundamental open problem in the design of wireless multi-hop networks is to efficiently schedule transmissions in a distributed manner. Implementing optimal scheduling involves solving a max-weight problem with secondary interference constraints which is provably NP-hard [1]. So researchers have focussed on proposing efficient approximation algorithms like maximal and greedy maximal scheduling, see [2]–[4] and references therein.

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, [5] propose random access schemes to approximate greedy maximal scheduling. Further, CSMA-CA can be viewed as a mechanism to get maximal schedules [3], 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 sub-optimal 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 work 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. Further, they choose among the rest of the schedules randomly, which benefits from the well-known performance traits of randomized algorithms [6]. Thus, the end result turns out to be quite good.

A. Contributions

We formally derive the worst case throughput ratio of CSMA-CA and optimal scheduling achieved in multi-hop networks, and then compare this ratio to the throughput ratio of maximal and optimal scheduling derived in [3]. CSMA-CA is a random access algorithm which 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 sub-optimal scheduling. This leads to a loss in throughput. Thus one may expect that CSMA-CA will have a worse performance.

In multi-hop topologies which do not have neighborhoods larger than 20 edges, for a max-min fair rate allocation, we analytically establish the following: (i) CSMA-CA achieves more than 16% of the optimal throughput always, (ii) 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 which 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 provably achieves between $1/3$ and $1/4$ of the optimal [4], [7], which is comparable to what CSMA-CA achieves.

In the process of formally deriving worst-case bounds for CSMA-CA, we also characterize the worst case 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 non-interfering 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.

II. Methodology

The objective of this paper is to study the worst case performance of CSMA-CA against optimal scheduling in multi-hop networks. CSMA-CA is a complex protocol and analyzing all aspects of the algorithm for all possible topologies one can construct is not analytically feasible. To reduce complexity, unlike prior works which either ignore certain aspects of the protocol or ignore real physical effects like collisions [8], [9], we choose to retain CSMA-CA as it is and simplify the analysis by restricting the topologies considered. However, the way we restrict topology is such that the scope of the problem is not reduced.

Problem Statement: We characterize the *neighborhood topology* which minimizes the ratio of the throughput achieved by CSMA-CA over that achieved by the optimal at the edge of interest at the *max-min rate allocation*.

A. Neighborhood Topology

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 $e_c \notin N_{e_c}$. We assume that the arrival process for each flow has i.i.d (independent and identically distributed) inter-arrival times. We further assume independence between the arrival process for different flows.

Instead of deriving worst case bounds over all possible multi-hop topologies and traffic matrices, we simplify our problem by studying neighborhood topologies only. Note that the traffic matrix in neighborhood topologies will obviously have only single-hop flows over each edge. This simplification does not come at the cost of reducing the scope of the objective. Studying neighborhood topologies suffices because there is a direct one-to-one connection in the throughput performance of neighborhood topologies and multi-hop topologies - The throughput performance of any multi-hop 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 multi-hop topology is dictated by its congested links [10]–[12]. For wireless networks, the idea of congested links gets replaced by congested neighborhoods.

Now lets try and understand what does the term congested neighborhood imply. To explain, we first define when two edges are said to interfere with each other, then define the term neighborhood, and finally explain the term congested neighborhood. Let T_e and R_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 T_{e_1} interferes with either T_{e_2} or R_{e_2}, or R_{e_1} interferes with either T_{e_2} or R_{e_2}. A neighborhood of an edge e is defined to be the set of edges which interfere with e.

Now, we explain the term congested neighborhood. In a multi-hop topology, for any rate allocation which operates at the boundary of the capacity region, each end-to-end flow passes through the neighborhood of at least one edge whose queue is fully utilized. These edges are called congested edges, and their neighborhoods are termed congested neighborhoods. (In our definition of neighborhood topologies, the congested edge is the edge of interest.)

Each end-to-end flow can potentially pass through a lot of congested neighborhoods, however we conjecture that the throughput achieved by each end-to-end flow is dictated by the throughput achieved in the most congested neighborhood it passes through. To establish this conjecture, we do extensive simulations following the approach taken by the Internet community to establish a similar conjectured relationship between performance and congested edges, see, for example [10]–[12].

B. Max-Min 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 $y_s > x_s$, then there must exist some s' such that $x_{s'} \leq x_s$ and $y_{s'} < x_{s'}$ [13].

To compare the throughput at the congested edge, we have to pick up a rate allocation point to compare. Amongst 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 it yields the worst throughput ratio at edge e_c among the three. Intuitively, more collisions are observed at the max-min allocation which results in a lower throughput with CSMA-CA. 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.

C. Model and Notations

This section introduces our notation and assumptions. 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 back-off window and the number of exponential back-off 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. To ensure that the difference between optimal scheduling and CSMA-CA is only due to the scheduling inefficiencies of CSMA-CA, we make the overhead imposed by control message exchanges and protocol headers to be the same for both. (Note that we expect the actual overhead required to implement optimal scheduling to be much higher, hence, the comparison is geared to favor optimal scheduling which makes the results obtained for CSMA-CA even stronger.)

All the results presented in this paper use the analytical model introduced in our prior work [14]. Due to space constraints, we do not present the model here. We merely re-state the concepts and notations which will be used in this paper while deriving the worst-case topology.

Four different two-edge topologies may exist in a multi-hop network [14]. (i) Coordinated Stations, (ii) Near Hidden Edges, (iii) Asymmetric Topology, and (iv) Far Hidden Edges. These two-edge topologies describe how different edges may interfere with each other.

Transmission on any of the neighboring edges which will cause the channel to be sensed busy by T_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 which cannot be sensed by T_e but will cause a collision at R_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 back-off window value at T_e. Let $p^e_{c,i}, 0 \leq i \leq m$ denote the RTS collision probability when the back-off window value at T_e is W_i. Note that $p^e_{c,i}, i \geq 1$ is a deterministic function of $p^e_{c,0}$. So, we adopt the convention that whenever we refer to the RTS collision probability from now on, we imply $p^e_{c,0}$.

D. Topology Characterization

In this section, we define a metric which will be used to characterize the worst case neighborhood. In the following definition, a time slot is equal to the transmission time of one packet.

Definition 2.1 (Interference Factor): The Interference Factor is the minimum number of time slots optimal scheduling will require to schedule all edges in N_{e_c} at least once.
The interference factor can also be viewed as the minimum number of maximal independent sets required to cover all the edges in N_{e_c}. We denote the interference factor of the set of edges N_{e_c} by $k_{N_{e_c}}$.

Theorem 2.1: Optimal scheduling yields a throughput of $\left(\left(k_{N_{e_c}}+1\right)T_s\right)^{-1}$ at edge e_c at the max-min rate allocation.

III. Worst Case Neighborhood

In this section, we give an $O\left(|N_{e_c}|^3\right)$ algorithm to determine the worst case neighborhood topology.

The performance of a given neighborhood topology depends on the following two interference characteristics. (i) $\forall e_i \in N_{e_c}$, 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. (ii) $\forall e_i, e_j \in N_{e_c}, i \neq j$, 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 [14] or optimal scheduling [15].

A. The Algorithm

Before presenting the algorithm, we first define two new variables which will be used to describe the algorithm. Let $k^{e_c}_{busy}$ and $k^{e_c}_{coll}$ denote the interference factor for the edges in $N^{e_c}_{busy}$ and $N^{e_c}_{coll}$ respectively. If none of the edges in $N^{e_c}_{busy}$ ($N^{e_c}_{coll}$) interfere with each other, then $k^{e_c}_{busy}=1$ ($k^{e_c}_{coll}=1$) ; and if all the edges in $N^{e_c}_{busy}$ ($N^{e_c}_{coll}$) interfere with each other, then $k^{e_c}_{busy}=|N^{e_c}_{busy}|$ ($k^{e_c}_{coll}=|N^{e_c}_{coll}|$).

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

$$TR_{worst} = \min_{|N^{e_c}_{coll}|}\min_{k^{e_c}_{busy}, k^{e_c}_{coll}}\left(TR_{worst}(|N^{e_c}_{coll}|, k^{e_c}_{busy}, k^{e_c}_{coll})\right) \quad (1)$$

The second part of the algorithm determines the value of $TR_{worst}(|N^{e_c}_{coll}|, k^{e_c}_{busy}, k^{e_c}_{coll})$.

1) Determining $TR_{worst}(|N^{e_c}_{coll}|, k^{e_c}_{busy}, k^{e_c}_{coll})$: To determine $TR_{worst}(|N^{e_c}_{coll}|, k^{e_c}_{busy}, k^{e_c}_{coll})$, we have to derive the two interference characteristics stated at the start of the section. 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 $N^{e_c}_{busy}$ interfere as near hidden edges with e_c.
(b) All edges in $N^{e_c}_{coll}$ interfere as asymmetric edges with e_c.
By definition of $N^{e_c}_{coll}$, between $e_i \in N^{e_c}_{coll}$ and e_c, e_c is the edge which is not aware of the channel state.

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: (i) We first determine how many edge pairs with one edge lying in $N^{e_c}_{busy}$ and the other lying in $N^{e_c}_{coll}$ interfere with each other. (ii) We next determine which of the $e_i, e_j \in N^{e_c}_{busy}, i \neq j$ interfere with each other. (iii) Finally, we determine which of the $e_i, e_j \in N^{e_c}_{coll}, i \neq j$ interfere with each other.

We first state a theorem to determine how many edge pairs with one edge lying in $N^{e_c}_{busy}$ and the other lying in $N^{e_c}_{coll}$ interfere with each other.

Theorem 3.2: The worst case neighborhood topology has the following properties.
(a) No edge in $N^{e_c}_{busy}$ interferes with any edge in $N^{e_c}_{coll}$.
(b) The throughput achieved by optimal scheduling is equal to $\left(\left(\max\ \left(k^{e_c}_{busy}, k^{e_c}_{coll}\right)+1\right)T_s\right)^{-1}$.

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

Theorem 3.3: The worst case neighborhood topology has the following properties.
(a) Two edges belonging to different maximal independent sets $I_j, I_k \in \mathcal{S}_I^{N^{e_c}_{busy}}, j \neq k$, interfere with each other.
(b) An edge in $N^{e_c}_{busy}$ will be contained in only one maximal independent set in $\mathcal{S}_I^{N^{e_c}_{busy}}$.
(c) $|I_j| = |N^{e_c}_{busy}|/k^{e_c}_{busy}, \forall I_j \in \mathcal{S}_I^{N^{e_c}_{busy}}$.
(d) $p^{e_c}_{busy} \leq k^{e_c}_{busy}\left(1-(1-\lambda_{e_c}T_s)^{|N^{e_c}_{busy}|/k^{e_c}_{busy}}\right)$ where λ_{e_c} is the packet arrival rate at edge e_c.

The theorem implies that distributing the $|N^{e_c}_{busy}|$ edges uniformly amongst the $k^{e_c}_{busy}$ independent sets minimizes the throughput ratio. Theorem 3.3 defines the topology precisely only when $|N^{e_c}_{busy}|/k^{e_c}_{busy}$ is an integer. Otherwise, Theorem 3.3(d) can be used to derive a lower bound on the value of $TR_{worst}(|N^{e_c}_{coll}|, k^{e_c}_{busy}, k^{e_c}_{coll})$.

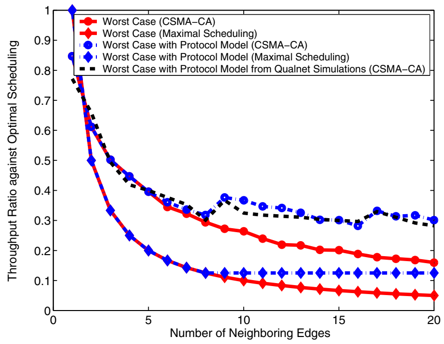


Fig. 1. Worst case and average throughput ratios for CSMA-CA and maximal scheduling against optimal scheduling for different neighborhood sizes.

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

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

(a) Two edges belonging to different maximal independent sets $I_j, I_k \in \mathcal{S}_I^{N^{e_c}_{coll}}, j \neq k$, interfere with each other.

(b) An edge in $N^{e_c}_{coll}$ will be contained in only one maximal independent set in $\mathcal{S}_I^{N^{e_c}_{coll}}$.

(c) $|I_j| = |N^{e_c}_{coll}|/k^{e_c}_{coll}, \forall I_j \in \mathcal{S}_I^{N^{e_c}_{coll}}$.

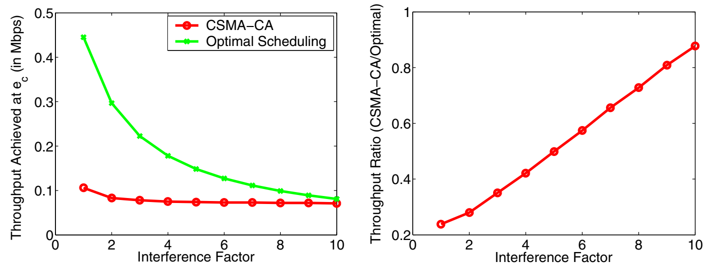
(d) $p^{e_c}_{c,0} \leq k^{e_c}_{coll}\left(1 - (1-\lambda_{e_c}T_s)^{|N^{e_c}_{coll}|/k^{e_c}_{coll}}\right)$ where λ_{e_c} is the packet arrival rate at edge e_c.

Recall that $p^{e_c}_{c,0}$ denotes the RTS collision probability. Similar to Theorem 3.3, Theorem 3.4 is exact only when $|N^{e_c}_{coll}|/k^{e_c}_{coll}$ is an integer, otherwise Theorem 3.4(d) can be used to derive a lower bound on $TR_{worst}(|N^{e_c}_{coll}|, k^{e_c}_{busy}, k^{e_c}_{coll})$.

Using the upper bounds on $p^{e_c}_{busy}$ and $p^{e_c}_{c,0}$ from Theorems 3.3 and 3.4, we derive a lower bound on $TR_{worst}(|N^{e_c}_{coll}|, k^{e_c}_{busy}, k^{e_c}_{coll})$. Applying these bounds to evaluate Equation (1), we find that the lower bound on TR_{worst} always occurred when $|N^{e_c}_{busy}|/k^{e_c}_{busy}$ and $|N^{e_c}_{coll}|/k^{e_c}_{coll}$ were integers. Recall that the bound is exact for these conditions.

B. Numerical Results

The red solid line with dots in Figure 1 plots the worst case throughput ratio as a function of the number of neighboring edges ($|N_{e_c}|$) for the system parameters summarized in Table I. Note that we use the default parameters of CSMA-CA, and do not optimize on the parameters to improve the throughput. (Unless explicitly stated, all the numerical results presented in this paper assume the system parameters of Table I.) 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 till $N_{e_c} = 20$. The ratio keeps on decreasing as $|N_{e_c}|$ increases, and it is slightly larger than 16% for $|N_{e_c}| = 20$. Figure 1 also plots the worst case throughput ratio of maximal scheduling (red solid line with rhombus) and we observe that CSMA-CA significantly outperforms maximal scheduling. (The other plots in this figure will be derived in Section IV.)



(a) (b)

Fig. 2. (a) Worst case throughput achieved at e_c as a function of the interference factor for $|N_{e_c}| = 10$. (b) Worst case throughput ratio achieved at e_c as a function of the interference factor for $|N_{e_c}| = 10$.

Packet Payload	1024 bytes
MAC Header	34 bytes
PHY Header	16 bytes
ACK	14 bytes + PHY header
RTS	20 bytes + PHY header
CTS	14 bytes + PHY header
Channel Bit Rate	1 Mbps
Propagation Delay	1 μs
Slot Time	20 μs
SIFS	10 μs
DIFS	50 μs
Initial back-off window	31
Number of exponential back-off windows	5
Auto-Rate Adaption	Off
Buffer Size and Maximum Retry Limit	Very Large

TABLE I
SYSTEM PARAMETERS USED TO OBTAIN NUMERICAL RESULTS.

Now, we present the value of $|N^{e_c}_{coll}|$ and the interference factor as determined by the first part of the algorithm for the worst case topology. $|N^{e_c}_{coll}|$ for the worst case topology is always equal to $|N_{e_c}|$. This is not surprising as collisions cause exponential back-offs reducing the throughput drastically. The interference factor for the worst case topology is always equal to 1 irrespective of the value of $|N_{e_c}|$. 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 non-interfering neighboring edges independently rather than simultaneously. Figures 2(a) and 2(b) show 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.

IV. 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 - (i) it establishes absolute worst case bounds, and (ii) it gives an analytical explanation of the surprising observation that the throughput ratio of CSMA-CA and optimal improves as the interference factor increases.

In addition to the assumptions specified in Section II, 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 [3], [4]. To be able to make a fair comparison as well as maintain analytical tractability, we also use the protocol model of interference.

In this section, instead of setting the interference range so as to get more realistic results with the protocol model, 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. So 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 bounds the maximum number of non-interfering neighboring edges of e_c. Using a derivation similar to [3], we derive that the maximum number of nodes which interfere with either T_{e_c} or R_{e_c} but do not interfere with each other is equal to 8.

A. 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, evaluating the performance of CSMA-CA and optimal scheduling for each of these topologies using the analytical model and finding the one which has the worst throughput ratio. This procedure formally establishes the worst-case bounds for CSMA-CA with the protocol model.

The blue dashed line with dots in Figure 1 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 non-interfering 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. Figure 1 also plots the worst case throughput ratio of maximal scheduling (blue dashed line with rhombus) which is equal to 12.5%.

The worst case topology is a by-product of this exercise, but we do not present the characteristics of the worst-case topology due to space constraints and focus only on the comparison of worst case bounds for CSMA-CA and maximal scheduling. For completeness, without going into details, we merely point out that an important observation made in Section III that the worst case topology has the minimum possible interference factor still holds.

V. Conclusions

This paper formally establishes worst-case throughput performance bounds for CSMA-CA in multi-hop networks. We observe that under similar assumptions, CSMA-CA easily outperforms maximal scheduling and achieves worst-case performance close to greedy maximal scheduling. The results presented in this paper have the following two implications. (i) As stated before, it motivates the use of random access schedulers. (ii) Secondly, it prompts researchers to investigate the design of practical congestion control and rate allocation protocols which can realize this good performance over random access schemes.

References

[1] G. Sharma, R. Mazumdar, and N. Shroff, "On the complexity of scheduling in wireless networks," in *Proceedings of ACM MOBICOM*, 2006.

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

[3] P. Chaporkar, K. Kar, X. Luo, and S. Sarkar, "Throughput and Fairness Guarantees Through Maximal Scheduling in Wireless Networks," *IEEE Transactions on Information Theory*, 2008.

[4] C. Joo, X. Lin, and N. Shroff, "Understanding the capacity region of the greedy maximal scheduling algorithm in multi-hop wireless networks," in *Proceedings of IEEE INFOCOM*, 2008.

[5] C. Joo and N. Shroff, "Performance of random access scheduling schemes in multi-hop wireless networks," in *Proceedings of IEEE INFOCOM*, 2007.

[6] R. Motwani and P. Raghavan, *Randomized Algorithms*, 1st ed. Cambridge University Press, 1995.

[7] M. Leconte, J. Ni, and R. Srikant, "Improved bounds on the throughput efficiency of greedy maximal scheduling in wireless networks," in *Proceedings of ACM MOBIHOC*, 2009.

[8] D. Chafekar, D. Levin, V. A. Kumar, M. Marathe, S. Parthasarathy, and A. Srinivasan, "Capacity of asynchronous random-access scheduling in wireless networks," in *Proceedings of IEEE INFOCOM*, 2008.

[9] L. Jiang and J. Walrand, "A distributed CSMA algorithm for throughput and utility maximization in wireless networks," in *Proceedings of the 46th Annual Allerton Conference on Communication, Control and Computing*, 2008.

[10] S. Floyd, "Connections with multiple congested gateways in packet-switched networks part 1: one-way traffic," *ACM SIGCOMM Computer Communication Review*, vol. 21, no. 5, pp. 30–47, 1991.

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

[12] F. Papadopoulos, K. Psounis, and R. Govindan, "Performance preserving topological downscaling of internet-like networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 12, 2006.

[13] D. P. Bertsekas and R. G. Gallagher, *Data Networks*, 2nd ed. Prentice Hall.

[14] A. Jindal and K. Psounis, "The Achievable Rate Region of 802.11-Scheduled Multihop Networks," *IEEE/ACM Transactions on Networking*, vol. 17, no. 4, pp. 1118–1131, 2009.

[15] K. Jain, J. Padhye, V. Padmanabhan, and L. Qiu, "Impact of interference on multi-hop wireless network performance," in *Proceedings of ACM MOBICOM*, 2003.