Synchronized CSMA Contention: Model, Implementation, and Evaluation

Ehsan Aryafar, *Member, IEEE*, Theodoros Salonidis, *Member, IEEE, ACM*, Jingpu Shi, *Member, IEEE*, and Edward Knightly, *Fellow, IEEE*

***Abstract*—A class of carrier sense multiple access (CSMA) protocols used in a broad range of wireless applications uses synchronized contention where nodes periodically contend at intervals of fixed duration. While several models exist for asynchronous CSMA contention used in protocols like IEEE 802.11 MAC, no model exists for synchronized CSMA contention that also incorporates realistic factors like clock drifts. In this paper, we introduce a model that quantifies the interplay of clock drifts with contention window size, control packet size, and carrier sense regulated by usage of guard time. Using a field programmable gate array (FPGA)-based MAC protocol implementation and controlled experiments on a wireless testbed, we evaluate the model predictions on the isolated and combined impact of these key performance factors to per-flow throughput and fairness properties in both single-hop and multihop networks. Our model and experimental evaluation reveal conditions on protocol parameters under which the throughput of certain flows can exponentially decrease; while at the same time, it enables solutions that can offset such problems in a predictable manner.**

***Index Terms*—Clock drift, contention window, CSMA, guard time, medium access control, multihop wireless networks, synchronized contention, topological bias.**

I. Introduction

SYNCHRONIZED carrier sense multiple access (S-CSMA) protocols partition time into periodic cycles of fixed duration. At the beginning of each cycle, nodes contend on a common channel using carrier sense and control packet handshake similar to RTS/CTS; after winning contention, nodes transmit until the end of the cycle. Synchronized CSMA contention has been used in both single-hop and multihop wireless networks for a wide variety of tasks. In single-hop networks, the IEEE 802.11 WLAN standard [1] supports synchronized contention in power-saving mode such that hosts wake up periodically and contend within a short window using beacons. In multihop networks, synchronized contention has been used for channel selection [2], [3] and per-channel contention [4] in multichannel networks, for leader election and power saving in sensor networks [5]–[7], and for antenna selection in multiple-input–multiple-output (MIMO) networks [8], [9].

In practice, clock drift is an integral part of any multihop synchronized network and cannot be avoided. The amount of clock drift can range from several milliseconds [10] to several microseconds [11], [12] and is dependent on the specific clock synchronization mechanism, the level of message exchange among the nodes, and the number of nodes/hops in the network. Drift introduces a clock phase bias in synchronized CSMA protocols because flows starting contention earlier than others may have higher chances to win the medium. In addition, multihop systems introduce topological bias where some flows compete with asymmetric views of channel state. To the best of our knowledge, this paper is the first to systematically and comprehensively investigate via modeling and experimental evaluation the joint impact of MAC protocol parameters and both types of bias on the throughput performance and fairness properties of synchronized CSMA systems. Our contributions are as follows.

First, we introduce and implement S-CSMA, a protocol that captures the basic features of synchronized CSMA contention. Our implementation on a field programmable gate array (FPGA) wireless platform enables realistic evaluation of S-CSMA and fine-grained control of the main factors that affect its fairness properties: *contention window size*, *control packet size*, *clock drift* (that shows the degree of imperfect synchronization), and *guard time* (an idle time duration at the end of each flow's cycle that controls the impact of carrier sensing in the presence of clock drifts) at the end of each transmission cycle.

Second, we introduce a Markovian model for throughput prediction in single-hop S-CSMA networks (e.g., synchronized WLANs) that incorporates clock phase drifts. Existing Markovian models for *asynchronous* CSMA protocols such as 802.11 [13]–[19] can model stochastic contention instants due to carrier sense, but are not suitable for periodic contention. Our model is different in nature and exploits the periodic structure of synchronized contention. Alternatively, Markovian approaches have also been used to model the throughput and delay of MAC protocols in single-hop synchronized sensor networks [20]–[22]. However, these models do not incorporate clock drifts, control packet size, and guard time, which we show to be critical factors in a practical implementation. Using the model and experimental evaluation on an FPGA platform, we show that flows with the earliest (latest) clock phases receive maximum (minimum) throughput and that the throughput of latest flows decreases sharply with clock phase drift. Surprisingly, these phenomena occur irrespective of guard-time usage.

Manuscript received November 27, 2011; revised August 12, 2012; accepted September 22, 2012; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor E. Modiano. Date of publication December 11, 2012; date of current version October 11, 2013. This work was supported by the NSF under Grants CNS-0751173 and CNS-0721894.

E. Aryafar is with the Department of Electrical Engineering, Princeton University, Princeton, NJ 08544 USA (e-mail: earyafar@princeton.edu).

T. Salonidis is the IBM T. J. Watson Research Center, Yorktown Heights, NY 10598 USA (e-mail: tsaloni@us.ibm.com).

J. Shi is with Doliver Capital Advisors L.P., Houston, TX 77005 USA (e-mail: jingpushi@yahoo.com).

E. Knightly is with the Department of Electrical and Computer Engineering, Rice University, Houston, TX 77005 USA (e-mail: knightly@rice.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.2012.2228225

Third, we consider multihop networks where in addition to their clock phase differences, flows compete with an asymmetric view of channel state and can suffer from lack of transmission opportunities due to carrier sensing, or high collision probability of their control packets due to hidden terminals. We extend our Markovian model to basic representative multihop scenarios. Our model and experimental testbed evaluation show that due to asymmetric channel views (i.e., topological bias) in multihop networks, flows with the earliest clocks do not necessarily receive the highest throughput. Instead, they may even receive zero throughput under certain conditions that we identify. For both guard-time and no-guard-time systems, we derive simple relationships among contention window sizes and clock phases that guarantee starvation-free operation. According to these relationships, in no-guard-time systems, each flow requires clock phase knowledge of two-hop neighbors; while in guard-time systems, one-hop clock phase knowledge is sufficient. This implies different requirements on the phase bounds provided by the clock synchronization mechanism running in the network.

Fourth, we consider arbitrary topologies and introduce an approximate model that provides a lower bound on the guard-time system throughput. This model decouples the joint effect of interfering flows and allows to express the throughput of each flow as a simple function of its own contention window size and the *harmonic* mean of the contention window sizes of its one-hop interferers. Using the model and experiments we show that: 1) the throughput of a flow with a topology disadvantage *exponentially* decreases with control packet size of its one-hop interferers; 2) the disadvantage of such a flow can only be offset by making one-hop interfering flows less aggressive through increasing the *harmonic mean* of their contention windows; and 3) global fairness objectives such as max-min fairness may require large contention window sizes for all flows, which can lead to increased network delays.

Finally, we discuss the application of our model to design contention window adjustment mechanisms that address unfairness in S-CSMA networks due to clock drifts and topological (dis)advantages. Unfairness in synchronized networks has been addressed in the context of slotted ALOHA systems (e.g., [23]). However, these models assume perfect synchronization and do not incorporate carrier sensing and control packets for medium access reservation, which we show to be critical factors in S-CSMA networks. In our proposed solution, we first identify minimum reference rates for the network flows that reflect network congestion and can be achieved by appropriate contention window selection. We next propose a distributed contention window adjustment mechanism that achieves or exceeds these reference rates and significantly enhances fairness among the flows.

This paper is organized as follows. In Section II, we introduce and implement S-CSMA and describe our experimental methodology. In Section III, we introduce the Markovian model for single-hop networks and investigate the impact of imperfect synchronization. In Section IV, we extend this model to representative multihop scenarios and investigate the impact of carrier sense. In Section V, we model arbitrary topologies for guard-time systems and investigate the impact of interference on the resulting throughput approximation and the impact of fairness objectives on contention window adjustment mechanisms. In Section VI, we propose contention window adjustment mechanisms that address unfairness in S-CSMA networks. Section VII concludes the paper.

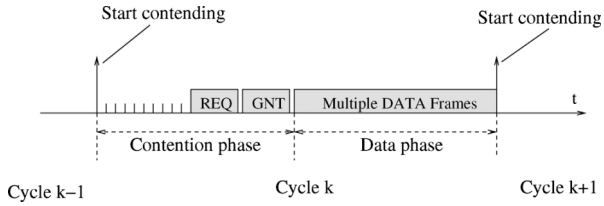


Fig. 1. Basic operations of S-CSMA.

II. S-CSMA

The detailed operations of synchronized CSMA protocols are all different from each other and, as a result, there is no single protocol model that characterizes all the operational details of these protocols. To analyze the basic nature of synchronized contention and capture its fundamental fairness properties, in this section we present a simple Synchronized CSMA protocol, which we call S-CSMA. Our protocol is not designed to capture all the details of this protocol family nor for improving performance. It is designed to be simple but characterize the basic use of synchronized contention. The principle and methodology of our analysis can be adapted to the analysis of any particular protocol.

A. Protocol Description

S-CSMA is a single-channel synchronized CSMA protocol where time is partitioned into fixed-duration cycles. As shown in Fig. 1, each cycle consists of a contention phase followed by a data transmission phase. At the beginning of each cycle, each node i starts contending for the channel by sensing the medium. Once the medium becomes idle, the node computes a random backoff counter based on an initial contention window W_i and starts counting down mini-slots while sensing the medium. If during this time a transmission is sensed, the node quits contention and waits for the next cycle. If the backoff timer expires, the node transmits a short request (REQ) control packet and waits for a short grant (GNT) control packet. If the GNT is received, the data phase begins *immediately*, and the node transmits data until the end of the cycle. The data transmission phase may consist of one or more data-link frames and their corresponding acknowledgments. If GNT is not received, the node doubles its contention window, computes a new random backoff counter, and contends again. The node will stop contending either when GNT is received or at the end of the contention phase. If the end of the contention phase is reached, the node resets its contention window to W_i, quits contention, and waits for the next contention cycle.

Since the node clocks are not perfectly synchronized, *leading* flows whose transmitters begin contention earlier at each cycle have a *phase advantage* over *lagging flows*. Carrier sensing is known to play a fundamental role in affecting the fairness properties of asynchronous CSMA protocols such as IEEE 802.11 [24]. In synchronized protocols, it is also an important factor that may either alleviate or aggravate the *phase bias* introduced by clock drifts.

In addition to phase bias, a multihop S-CSMA network exhibits *topological bias*. More specifically, a flow A has a topological advantage with respect to flow B if the transmitter of A is within range of the receiver of B but not its transmitter. Being a hidden terminal, the advantaged flow can be blocked only when its transmitter senses the GNT packet of the disadvantaged flow receiver. Thus, the disadvantaged flow can win contention only if it completes its random backoff countdown plus REQ packet transmission before the advantaged flow completes its random backoff countdown. The degree of topological bias is essentially determined by the duration of the REQ control packet.

A system design alternative is to employ a *guard time* at the end of the data phase. In such a guard-time system, upon finishing data transmission, each flow waits for an additional duration $T_{\rm gb}$ before starting contention for the next cycle. Since $T_{\rm gb}$ typically exceeds the maximum clock drift in the network, a guard-time system may avoid starvation problems due to carrier sensing transmissions of previous cycle. On the other hand, it may eliminate potential opportunities to alleviate the phase bias. Phase bias, topological bias, carrier sense, and the use of guard time are tightly coupled and can dramatically affect the fairness properties of multihop synchronized CSMA systems. In the next sections, we introduce models that treat such systems in a unified manner and address critical design issues such as: 1) impact of phase bias and topological bias to per-flow throughput; 2) effectiveness of carrier sense in alleviating phase bias; 3) requirements for starvation-free clock synchronization mechanisms,; and 4) guidelines for contention window adjustment mechanisms to address unfairness due to both phase and topological bias.

B. Protocol Implementation

We implemented the S-CSMA protocol on the WARP platform [25]. Our implementation has the following main components.

Synchronization: Our goal is to investigate the behavior of synchronized CSMA contention over a wide range of clock phase drifts. Hence, instead of implementing a particular wireless synchronization mechanism, we use an implementation that enables control of clock phase drifts at high granularity and over a wide range. In our implementation, one node is responsible for announcing the beginning of each cycle to the rest of the nodes in the network. This synchronizer node has several dedicated output debug header pins connected through wire to an input debug header pin of each other node. The synchronizer node generates pulse signals on its output header pins. The rest of the nodes continuously monitor their input header pins and create a high-priority interrupt whenever a rising edge is detected on their input header pins. This rising edge announces the beginning of the cycle. The synchronizer node determines the clock phase drift of each node by sending the corresponding pulse signals at different times. Our implementation approach allows to model the drift effect of any desired synchronization mechanism and control the clock phase drift on a per-node basis at 1 μs accuracy.

Carrier Sense: WARP supports timers driven by 40-MHz crystal oscillators within the FPGA fabric, which enable small mini-slot durations. We implemented S-CSMA with 20 μs mini-slot duration for carrier sensing, also used by IEEE 802.11b. In our implementation, a carrier-sense timer accepts a number of mini-slots and counts them down, provided that the medium is idle for the duration of mini-slot. If the medium becomes busy, the timer stops and only resumes counting down after a distributed interframe space (DIFS) period.

Cycle Elapsed Time: Each node has a timer that computes the elapsed time since the beginning of the cycle. This timer is used for: 1) regulating REQ/GNT transmissions during the contention phase; 2) checking if a packet fits the cycle prior to its transmission; and 3) accounting for guard-time duration in a guard-time system.

REQ/GNT Handshake: Before transmitting a REQ packet, the contention window is set, and the MAC counts down with the carrier-sense timer. When this timer expires, the node transmits the REQ packet.

Binary Exponential Backoff (BEB): If after a REQ transmission GNT is not received in a timeout, the sender doubles its contention window similar to the IEEE 802.11 BEB rule.

C. Experimental Methodology

We conduct our experiments in a nine-node in-lab testbed. Each node consists of a laptop connected through Ethernet to a WARP board. Each WARP board is also connected to a 3-dBi external antenna. The synchronizer node is connected to the rest of nodes using long low-voltage wires. The boards operate at 5-GHz band and 12-Mb/s data rate. Topology control is performed by adjusting the transmit power and node locations. WARP supports a wide range of transmission power levels where the gain in the transmit path of RF transceiver can be changed over a 60+ dB range. Still, it has been challenging to perfectly realize the desired multihop topologies due to the in-lab testbed and varying wireless channel conditions.

Unless otherwise specified, the experiments use 200-B data packets, 24-B REQ/GNT packets, 30 ms cycle length, and 5 ms contention phase. In a guard-time system, the guard time is set to 1 ms to account for the wide range of clock drifts investigated in the experiments. The laptops send backlogged UDP traffic using *iperf*. Each data point is an average of five experiments, each lasting for 200 s.

III. Modeling Single Hop Networks

We develop a discrete-time Markov chain model to predict the per-flow success probability in a single-hop S-CSMA network. We account for temporal (dis)advantages introduced by the different clock phases in our model, where the system state for a cycle represents which flow transmits during this cycle. The transition probability is determined by relative clock phases of the flows and other system factors. This allows to relate the clock phase to the stationary distribution of the system state. Using this model, we investigate the joint effects of imperfect synchronization, carrier sense, and guard time.

A. Model

We define a flow to be a transmitter–receiver pair. We use the terms contention window, backoff counter, and phase for flows instead of their transmitter nodes.

Let $t_i(k)$ be the time instant where the kth cycle of flow i begins. Let $T_{\rm c}$ and $T_{\rm d}$ denote the duration of the contention phase and the data transmission phase, respectively. The fixed duration of a cycle is then given by $T = T_{\rm c} + T_{\rm d} + T_{\rm gb}$, where $T_{\rm gb}$ is the duration of guard time. We characterize each flow

by its clock phase θ_i[1] with respect to an absolute global clock reference (or alternatively, the earliest clock in the network). Given the phase θ_i, the contention time instant of each flow i for cycle k is $\theta_i + kT$. We assume that the clock frequencies of all flows are equal and constant and that the clock phase difference between any two flows can be at most $\theta_{\max}$. $\theta_{\max}$ can be either known (as a maximum error of a clock synchronization mechanism) or unknown to the flows. We assume $\theta_{\max}$ is much smaller than T, the duration of a cycle, and is smaller than $T_{\rm gb}$ in a guard-time system. We also denote by $\theta_{ij} = \theta_i - \theta_j$ the relative phase between flows i and j. All of these quantities are expressed in terms of mini-slots. We assume that control packet losses are only due to collisions. Control packets are typically transmitted at lower data rates than data packets to achieve high robustness in wireless channel conditions.[2]

Performance Metric: In our model and experimental evaluation, we use per-flow success probability as throughput performance metric. Success probability is the time fraction a flow successfully reserves the channel at the beginning of each cycle. Since this metric captures fraction of transmission opportunities, it is independent of packet size, data rate, cycle duration, and guard-time duration.[3] We will use success probability and throughput interchangeably, both referring to the same metric. Data packet losses due to channel conditions can be incorporated in our model by multiplying the calculated throughput with data packet success probability.

System State: Consider a fixed number N of contending flows, indexed by $1, 2, \ldots, N$, respectively. We let $b(k)$ denote the index of the flow accessing the channel at cycle k. We model the evolution of stochastic process $b(k)$ by a discrete-time Markov chain, in which the state of the system in cycle k is $b(k)$, and the state space is $S = \{1, 2, \ldots, N\}$. The transition probability from state i to state j, denoted by $p_{ij} = P\{b(k) = j | b(k-1) = i\}$, is the probability flow j wins contention at cycle k given that flow i transmitted during cycle $k-1$. Note that the transition probabilities do not depend on k because in a synchronized CSMA system the flows refresh their contention state at the beginning of each cycle. Solving the Markov chain, we obtain the stationary distribution $\boldsymbol{\pi} = \{\pi_i\}$, $\forall i \in S$, where π_i is the success probability of flow i.

Transition Probability: We now compute the transition probabilities p_{ij} for both guard-time and no-guard-time systems. We denote by X_i the random backoff counter computed by each flow i at a contention cycle and use $p_i(x)$ to denote $P(X_i = x)$ and $\Phi_i(x)$ to denote $P(X_i > x)$.

No-Guard-Time System: Let i be the flow transmitting during cycle $k-1$. We divide all other flows into two sets, the leading set $\mathrm{LD}(i) = \{m : \theta_m \le \theta_i\}$ and the lagging set $\mathrm{LG}(i) = \{m : \theta_m > \theta_i\}$. At the beginning of cycle k, flow i will sense idle medium and begin random backoff at its contention instant $t_i(k)$. Any flow m in leading set $\mathrm{LD}(i)$ will also begin backoff at $t_i(k)$ because it will sense data transmissions of flow i during cycle $k-1$. On the other hand, any flow m in lagging set $\mathrm{LG}(i)$ will start backoff at its later contention instant $t_m(k)$, provided neither flow i nor its leading flows have counted down their backoff counters to zero by that time instant.

Now, let j be the flow that wins contention at cycle k. If flow j is in set $\{i\} \cup \mathrm{LD}(i)$, and its backoff counter $X_j = x$, it will win contention if: 1) all other flows in $\{i\} \cup \mathrm{LD}(i)$ use a random backoff counter greater than x; and 2) the random backoff counters of all flows in lagging set $\mathrm{LG}(i)$ expire after $t_i(k) + x$. On the other hand, if flow j is in the lagging set $\mathrm{LG}(i)$, it will win contention only if the random backoff counters of all other flows expire after $t_j(k) + x$. After taking expectation with respect to x, the transition probability p_{ij} is given by

$$p_{ij} = \begin{cases} \sum_{x=0}^{W_j-1} p_j(x) \prod_{m:\theta_m \le \theta_i, m \ne j} \Phi_m(x) \\ \quad \times \prod_{m:\theta_m > \theta_i} \Phi_m(\theta_{jm} + x), & \theta_{ji} \le 0 \\ \sum_{x=0}^{W_j-1} p_j(x) \prod_{m:\theta_m \le \theta_i} \Phi_m(\theta_{ji} + x) \\ \quad \times \prod_{m:\theta_m > \theta_i, m \ne j} \Phi_m(\theta_{jm} + x), & \theta_{ji} > 0. \end{cases} \quad (1)$$

Eq. (1) does not handle collisions, resulting in the sum of the transition probabilities out of a system state being slightly less than 1 ($\sum_j p_{ij} < 1$). To handle collisions, we add an additional collision state c to the system state space. The transition probability from state i to state c is $1 - \sum_j p_{ij}$. To compute p_{cj}, we assume the time it takes for flows to detect and handle collision at the beginning of the cycle is larger than the clock phase differences. Since clock phases are absorbed, we assume that each flow has equal access opportunity after a collision, i.e., p_{cj} is equal for all j. This simplifying assumption does not incorporate the impact of increased contention window size of the colliding flows. However, since flows reset their contention windows at the beginning of each cycle, it does not compromise the model's accuracy. This is also verified through our experimental evaluations in Section III-B.

Guard-Time System: When guard time is present, no node will sense a busy channel at its contention instant of cycle k and will immediately start random backoff. Since carrier sense at the beginning of each cycle has no effect, the transition probabilities p_{ij} are independent of which node i transmitted during the previous cycle. Therefore, the transition probabilities p_{ij} for the guard-time system are given by

$$p_{ij} = \sum_{x=0}^{W_j-1} p_j(x) \prod_{m:m \ne j} \Phi_m(\theta_{jm} + x). \quad (2)$$

Collisions are handled by adding a collision state, similar to the no-guard-time system.

B. Clock Drift Experimental Investigation

We create a four-flow single-hop network in our testbed where all transmitters are within sensing range. Each flow i uses a contention window of $W_i = 32$ mini-slots and draws its backoff counter uniformly within this window in each contention cycle. The clock phases are fixed to $\theta_i = i \times 10$ mini-slots, $i = 0, \ldots, 3$.

We first consider a no-guard-time system. Fig. 2(a) depicts the measured and predicted throughput as a function of the flow clock phases sorted in nondecreasing order. We observe that the throughput falls off sharply with the phase distance from the

[1]Clock phase captures the impact of nodes starting contention at different times due to imperfect clock synchronization and propagation delay.

[2]In hostile environments with extremely adverse channel conditions, control packets can be lost even if transmitted at the lowest data rate. Modeling these issues is an interesting topic for future work.

[3]These parameters are fixed during the protocol operation and can be used to determine throughput in megabits per second.

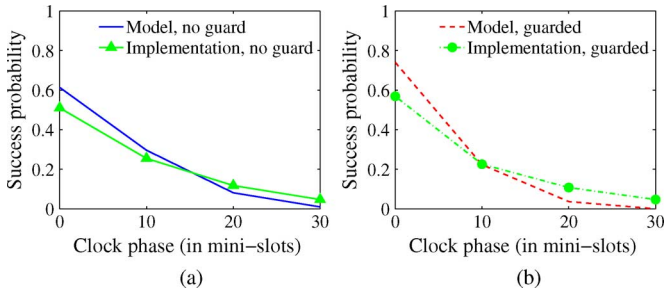


Fig. 2. Model predictions and testbed measurements for a four-flow single-hop network using S-CSMA. (a) No guard time. (b) Guard time.

earliest clock, which can be explained by our model: Eq. (1) predicts that when the clock phase of flow j becomes larger, p_{ij} becomes smaller, and at the same time p_{ji} becomes larger for all i, thus rapidly decreasing the stationary probability of state j. The guard-time system, depicted in Fig. 2(b), follows a similar trend as the no-guard-time system. Thus, different clock phases can lead to unfairness or even starvation, regardless of guard-time usage.

Overall, the model provides a very good prediction of the experimental results. However, we observe that the model slightly overestimates the throughput of the earlier flows, while underestimating the throughput of the later flows. After a close inspection of the packet traces, we identified control packet collisions as the main reason of the model inaccuracy. In particular, we observed several control packet collisions among the earlier flows, which reduced their success probability, while providing transmission opportunities for the later flows.

IV. Analyzing the Role of Carrier Sense in S-CSMA Multihop Networks

In a multihop S-CSMA network, flows contend with different channel state. This creates additional sources of unfairness or starvation. More specifically, the throughput of a flow can be very low due to either high collision probability or lack of transmission opportunities, or both. High collision probability arises when a flow transmitter is unable to sense the activity of out-of-range flows causing collisions at the flow receiver. Lack of transmission opportunities arises when a flow transmitter senses misaligned transmissions within its vicinity for extended time periods. These two sources of starvation are best illustrated by two representative scenarios, namely Information Asymmetry (IA) and Flow-in-the-Middle (FIM). In this section, we extend the analytical model in Section III to investigate each of these problems.

A. Lack of Transmission Opportunities

In CSMA wireless networks, carrier sense can yield unfairness or even starvation when flows sense uncoordinated transmissions in their neighborhood. A representative scenario is FIM [19], [24], [26], shown in Fig. 3. In asynchronous CSMA protocols like 802.11, the outer flows are not within sensing range of each other, and their transmissions are not time-aligned. The middle flow continuously defers due to carrier sensing and can only contend in the small intervals where both outer flows are jointly idle. The result is very few transmission opportunities and very low throughput for the middle flow, even in a collision-free FIM scenario like Fig. 3 where each receiver is

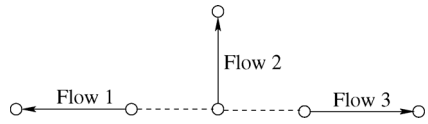


Fig. 3. FIM scenario.

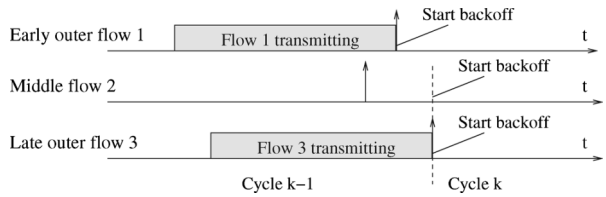


Fig. 4. Computation of p_{12}, the probability of middle flow 2 winning contention given the two outer flows transmitted in previous cycle. Arrows represent cycle boundaries.

only within range of its transmitter. In contrast to asynchronous CSMA, the middle flow throughput under S-CSMA can vary from zero to maximum depending on the relative clock phases of the flows and their interaction with carrier sense. We extend the model of Section III and use the FIM scenario to analyze these interactions and also evaluate the option of disabling carrier sense using guard time.

1) Model: The key observation that simplifies the analysis of the FIM scenario is that when one of the outer flows wins contention in a cycle, the other outer flow will also transmit in the same cycle, while the middle flow will not transmit. On the other hand, when the middle flow wins, both outer flows defer transmission during this cycle. We use two states in the model: state 1 for the two outer flows and state 2 for the middle flow.

Without loss of generality, we assume that flow 1 is the earlier outer flow ($\theta_1 \leq \theta_3$). We compute the transition probabilities p_{12} and p_{22} of middle flow 2 winning contention during cycle k, given that either the outer flows or the middle flow itself were transmitting during cycle $k-1$, respectively. Then, the other two transition probabilities can be determined as: $p_{11} = 1-p_{12}$ and $p_{21} = 1-p_{22}$. Given the transition probabilities, the success probability of the middle flow is given by the stationary probability closed-form expression of a two-state Markov chain

$$\pi_2 = \frac{p_{12}}{1 + p_{12} - p_{22}} \tag{3}$$

while the success probability of each of the outer flows is $\pi_1 = 1 - \pi_2$.

Transition Probability of Guard-Time System: When guard time is used, no flow senses busy carrier at the beginning of each cycle. Thus, in this case, the transition probabilities are computed similarly to the single-hop guard time case

$$p_{i2} = \sum_{x=0}^{W_2-1} p_2(x)\Phi_1(x+\theta_{21})\Phi_3(x+\theta_{23}), \qquad i = 1, 2. \tag{4}$$

Transition Probability of No-Guard-Time System: The transition probabilities depend on how the clock phase of the middle flow is related to the phases of the outer flows. We consider the case where the middle flow 2 is leading both outer flows, i.e., $\theta_2 \leq \theta_1 \leq \theta_3$. To compute p_{12}, the transition probability of the middle flow winning the contention in cycle k given the two outer flows transmitted in cycle $k-1$, we need to determine the time instants when the three flows start their backoff counters.

TABLE I
MODEL FOR FIM SCENARIO IN NO-GUARD-TIME SYSTEM: TRANSITION PROBABILITIES INTO STATE 2

	p_{12}	p_{22}
$\theta_2 \le \theta_1$	$\sum_{x=0}^{W_2-1} p_2(x)\Phi_1(x+\theta_{31})\Phi_3(x)$	$\sum_{x=0}^{W_2-1} p_2(x)\Phi_1(x+\theta_{21})\Phi_3(x+\theta_{23})$
$\theta_1 \le \theta_2 \le \theta_3$	$\sum_{x=0}^{W_2-1} p_2(x)\Phi_1(x+\theta_{31})\Phi_3(x)$	$\sum_{x=0}^{W_2-1} p_2(x)\Phi_1(x)\Phi_3(x+\theta_{23})$
$\theta_2 \ge \theta_3$	$\sum_{x=0}^{W_2-1} p_2(x)\Phi_1(x+\theta_{21})\Phi_3(x+\theta_{23})$	$\sum_{x=0}^{W_2-1} p_2(x)\Phi_1(x)\Phi_3(x)$

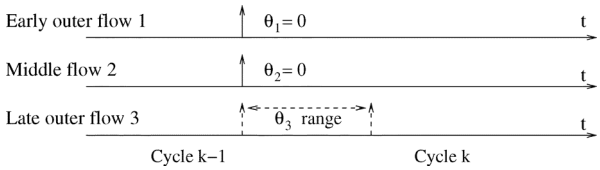


Fig. 5. Clock phase investigation: $\theta_2 = \theta_1 = 0, 0 \le \theta_3 \le 30$, all in mini-slots.

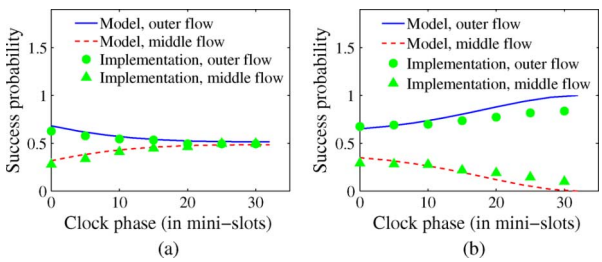


Fig. 6. FIM scenario, flow success probability as a function of θ_3. (a) Guard time; no guard time.

As shown in Fig. 4, the two outer flows sense an idle channel at their contention instants for cycle k and start their backoff counters immediately. However, due to carrier sense, flow 2 starts its backoff counter only when flow 3 finished its transmission for cycle $k-1$. Once the time instant for each flow to start its backoff counter is determined, the transition probability can be computed accordingly. The transition probabilities in different cases are summarized in Table I.

2) Clock Drift Experimental Investigation: According to the preceding analysis, the success probabilities depend on the relative phases of the three flows and whether guard time is used or not. In this section, we study the effect of these factors on success probability using the model predictions and experiments in our testbed. Unless otherwise specified, all flows use contention windows $W_1 = W_2 = W_3 = 32$ mini-slots.

Effect of Relative Phase of the Outer Flows θ_{31}: We set $\theta_2 = \theta_1 = 0$ and vary the phase θ_3 of the late outer flow 3 from 0 to 30 mini-slots, as shown in Fig. 5.

Guard-Time System: Fig. 6(a) shows a close match between measurements and model predictions. The middle flow 2 does not carrier sense at the beginning of each cycle, and the late outer flow 3 becomes less competitive as its lag from the middle flow 2 increases. Thus, the success probability of the middle flow π_2 increases and stabilizes after the clock phase difference between the two outer flows θ_{31} exceeds 32 mini-slots. After this point, flow 2 only competes with the early outer flow 1. On the other hand, flow 3 achieves equal throughput to flow 1, not due to its own contention effort, but because it transmits when flow 1 wins the contention.

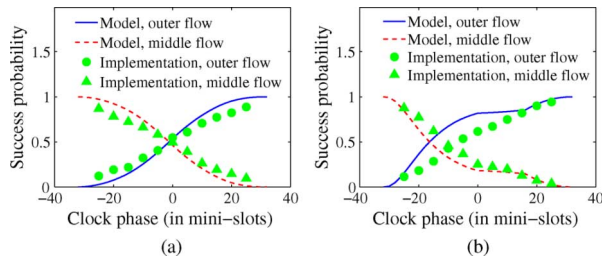


Fig. 7. FIM scenario, flow success probability as a function of θ_2. (a) Guard time. (b) No guard time.

No-Guard-Time System: Fig. 6(b) shows that the effect of θ_{31} on the middle flow is reversed when guard time is not used. As the phase θ_3 of the late outer flow 3 increases, the success probability of the middle flow decreases. This is because the middle flow would defer contention due to carrier sensing the data transmission of the late outer flow. At $\theta_3 > 32$ mini-slots, the middle flow receives zero throughput. In this case, flow 1 starts backoff immediately, but flow 2 carrier senses the transmission of flow 3 and delays its contention for an amount of time greater than the contention window of flow 1.

Effect of Middle Flow Phase θ_2: We now investigate the effect of phase θ_2 of the middle flow for a fixed relative phase θ_{31} of the outer flows. More specifically, contention windows for all flows are set to 32 mini-slots and $\theta_1 = 0$, $\theta_3 = 16$ mini-slots so that the middle flow starvation condition $(W_1 < \theta_{31})$ does not hold. The phase of the middle flow θ_2 varies from -30 to 30 mini-slots.

Guard-Time System: Fig. 7(a) shows that the success probability π_2 of middle flow 2 decreases as its phase θ_2 increases with a smooth curve. Indeed, when guard time is used, the two transition probabilities to state 2 are equal, i.e., $p_{12} = p_{22}$, which implies that $\pi_2 = p_{12}$. Therefore, the two product terms of π_2 that depend on θ_2 appear in the numerator, resulting in a smooth curve.

No-Guard-Time System: The success probabilities are depicted in Fig. 7(b). We observe that when the middle flow 2 is leading both outer flows by more than W_2 mini-slots, flow 2 receives maximum throughput, while the outer flows receive zero throughput. This is verified through experimental results of Fig. 7(b). Indeed, if the middle flow 2 transmits during cycle $k-1$, its backoff counter at cycle k will always expire earlier than the time instants when the outer flows start backoff. Hence, once the middle flow 2 wins contention at a cycle, it will continue to do so at every subsequent cycle. This is confirmed by Table I, where in this case, $p_{21} = 0$ and $p_{22} = 1$—state 2 of the Markov model is an absorbing state.

As the clock phase of the middle flow 2 increases, its success probability decreases rapidly to zero. This is because p_{12} decreases fast and p_{21} increases fast. When the phase of the middle flow 2 is within the phases of the outer flows, i.e., $\theta_1 \leq \theta_2 \leq \theta_3$, the success probability of flow 2 decreases slowly. This is because p_{22} (Table I) has only one product term that depends on θ_2 as opposed to having two terms in other cases. We call this region the *flat region*.

3) Summary and Discussion: Overall, the model provides very good predictions of the experimental results. In some cases, the model underestimated the success probability of the middle flow, especially in the no-guard-time system [Figs. 6(b) and 7(b)]. We could identify two reasons. First, wireless channel variations occasionally caused the outer flows to interfere with each other, which decreased their throughputs. Second, our model assumes that a winning flow fills the entire cycle with packets due to backlogged flows. However, in our implementation, a winning flow transmits data packets as long as the data packet and the corresponding acknowledgment fit within its data phase (i.e., the data packet and the ACK are received before the end of the data phase). A close inspection of the packet traces showed that the last packets of an outer flow occasionally did not fit the cycle. In such cases, the middle flow can potentially sense more idle medium during its contention phase, and thus increase its success probability.

Our analysis showed that in a no-guard-time system, a (middle) flow can starve even if its clock is the earliest among the clocks within its vicinity. This is because some of its neighbors (outer flows) with very late clocks can significantly delay the time instant this flow starts to contend, resulting in some other neighbors always finishing contention earlier than this flow. In general, a flow will starve if the maximum phase among uncoordinated flows ("outer flows") within its vicinity exceeds the contention window of the early outer flow 1 ($\theta_{31} > W_1$). When this condition holds, this (middle) flow will receive zero throughput if its own contention window W_2 is greater than its relative phase to the early outer flow θ_{12}.

To avoid the above starvation phenomena a contention window adjustment solution should ensure that each node's contention window always exceeds the relative clock phase of both its one-hop and two-hop neighbors. Viewed from another angle, this also imposes stricter requirements for clock synchronization protocols to bound the clock phase within a two-hop instead of one-hop neighborhood of each node.

When guard time is used, the relative phases also play a role in throughput degradation. However, starvation is caused only if a flow is leading or lagging excessively with respect to all its outer flows. In the first case, the flow causes starvation to the outer flows while in the second case it starves. This also holds in the no-guard-time system. Although the guard-time system does not inherently offer phase jitter protection, success probability does not fall off as sharply as in regions other than the flat region of the no-guard-time system.

B. High Collision Probability Analysis

The IA scenario shown in Fig. 8 captures the high collision probability that arises in asynchronous CSMA protocols due to hidden terminals and asymmetric view of the channel state. Here, flow 2 is out of range of the transmitter of flow 1, whereas the transmitter of flow 2 is within range of the receiver of flow 1.

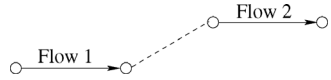


Fig. 8. IA scenario.

Thus, flow 1 cannot sense the activity of flow 2, while flow 2 can sense the activity of flow 1 by virtual carrier sensing the GNT (CTS) packets sent by the receiver of flow 1.

This asymmetry results in a disadvantage for flow 1 in an asynchronous CSMA protocol such as 802.11 [19], [24], [26]. Flow 1 randomly attempts to transmit the REQ (RTS) packet within the short backoff intervals of flow 2, possibly colliding with both short REQ packets and long data transmission. Thus, flow 1 repeatedly doubles its backoff contention window until the retransmission limit is reached, making it increasingly difficult for flow 1 to win the contention. In contrast to asynchronous CSMA, the disadvantaged flow's throughput under S-CSMA can vary from zero to maximum depending on the relative clock phases of the flows and their interaction with carrier sense. We extend the model of Section III and use the IA scenario to analyze these interactions and also evaluate the option of disabling carrier sense using guard time.

1) Model: We now use a two-state Markov chain model to analyze the IA scenario of Fig. 8 that captures the root cause of high collision probability. In this topology, flow 2 has complete information of the channel and has a topological advantage when contending with flow 1. We call flow 2 the advantaged flow and flow 1 the disadvantaged flow. In the model, system state 1 and 2 are for flow 1 and 2, respectively. We denote the duration of the REQ packet as R, and the duration of the GNT packet as G.

Perfect Synchronization: We first consider that the two flows are perfectly synchronized, i.e., $\theta_1 = \theta_2 = 0$. Due to topological difference between the two flows, for the REQ packet of the disadvantaged flow 1 to be successfully received, its entire transmission has to be finished before the backoff counter of the advantaged flow 2 expires. Therefore, the success probability of the disadvantaged flow 1 is given by

$$\pi_1 = p_{11} = \sum_{x=0}^{W_1-1} p_1(x)\Phi_2(x+R). \tag{5}$$

The success probability of the disadvantaged flow is given by $\pi_2 = 1 - \pi_1$.

Fig. 9 depicts the success probability of the two flows as a function of REQ packet duration[4] R. We observe that the success probability of the disadvantaged flow 1 decreases rapidly with R. When R is larger than the backoff window of the advantaged flow 2, the disadvantaged flow 1 receives zero throughout. This is because when $R > W_2$, $\Phi_2(x+R) = 0$ in (5). This indicates that even with perfect synchronization, flows can starve in multihop networks due to control packet size.

Imperfect Synchronization: We now analyze the impact of imperfect synchronization on the success probability of the two flows. We consider both guard-time and no-guard-time systems.

[4]REQ packet duration depends on the particular technology. For example, in 802.11b, an RTS packet is composed of a preamble (72 bits transmitted at 1 Mb/s), PLCP header (48 bits transmitted at 2 Mb/s), and MAC header (160 bits transmitted at 2 Mb/s), which adds up to 172 μs duration time or almost nine mini-slots (mini-slot duration in 802.11b equals 20 μs).

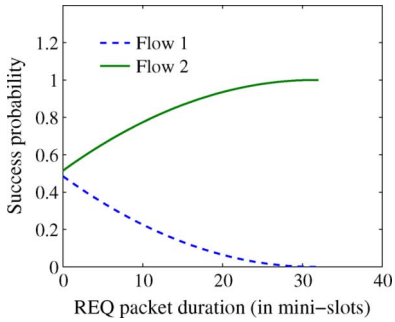


Fig. 9. IA: success probabilities as a function of REQ packet duration.

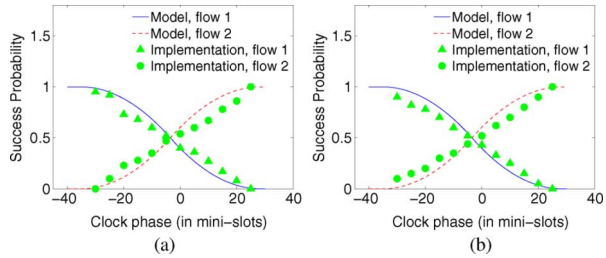


Fig. 10. IA scenario, flow success probability as a function of θ_1. (a) No guard time. (b) Guard time.

As in the FIM scenario, the transition probabilities of the model depend on the relative clock phases of the two flows. When guard time is used, none of the flows senses busy carrier at the beginning of each cycle. In this case, the success probability of the disadvantaged flow can be easily calculated from (5) by adding the relative phase difference between the two flows (θ_{12}) to the Φ_2 expression.

The analysis for the no-guard-time scenario is similar to that for the FIM scenario, except for computing p_{21}. To avoid redundancy, we only show how we derive p_{21}. When the disadvantaged flow 1 is leading the advantaged flow 2 ($\theta_1 < \theta_2$) in the no-guard-time system, the computation of the transition probability p_{21} becomes more complicated. If the advantaged flow 2 transmitted in cycle $k-1$, flow 1 will start backoff at cycle k before flow 2 ends its transmission for cycle $k-1$ because flow 1 cannot sense the transmission of flow 2. As a result, the receiver of flow 1 will not reply to the REQ packet due to either collision or virtual carrier sensing at this node. After a timeout, the transmitter of flow 1 doubles its backoff window and recontends according to the contention rule of S-CSMA. To model this process, we introduce random variable S_1 to denote the duration from $t_1(k)$, the contention time instant of flow 1 at cycle k, until flow 1 succeeds or the contention phase of its current cycle ends

$$S_1 = \sum_{s=0}^{N_1} (R + X_1^{(s)} + G) \tag{6}$$

where $X_1^{(s)}$ is the backoff duration at backoff stage s, and N_1 is the number of retries. Given S_1, the transition probability p_{21} can be computed as

$$p_{21} = \sum_{x_2=0}^{W_2-1} p_2(x_2)(\Phi_{S_1}(\theta_{21}) - \Phi_{S_1}(x_2+\theta_{21})), \qquad \theta_1 < \theta_2 \tag{7}$$

where $\Phi_{S_1}()$ is the complementary cumulative distribution function (cdf) of S_1. Derivation of $\Phi_{S_1}()$ in closed form is complicated because the terms $X_1^{(s)}$ follow different distributions and the number of retries N_1 is itself a random variable. Thus, we compute $\Phi_{S_1}()$ numerically using recursion on the number of backoff stages of flow 1.

2) Clock Drift Experimental Investigation: We investigate the evolution of success probabilities π_1 and π_2 as a function of θ_1, the phase of flow 1. The phase of flow 2 is set to zero, the REQ and GNT packet duration R an G are set to three mini-slots, and the contention windows W of both flows are set to 32 mini-slots.

The no-guard-time case is depicted in Fig. 10(a). For all values of θ_1, the success probabilities predicted by the model match the success probabilities obtained by experiments. Note that in many cases the success probabilities of the two flows are closer to each other than what is predicted by the model. Our investigation revealed that wireless channel variations occasionally caused the senders of the two flows to interfere with each other, which could potentially make the topology similar to a single-hop network and affect the predicted results.

When flow 1 is leading flow 2 by more than $W+R$ mini-slots ($\theta_{12} < -(W+R)$), flow 1 always wins contention and receives maximum throughput, while flow 2 receives zero throughput. After this point, the success probability of flow 1 decreases until it reaches zero when $\theta_1 > W - R$.

The curve for the guard-time system shown in Fig. 10(b) follows a similar trend to the curve of the no-guard-time system, with the same maximum, minimum, and crossover points. This confirms that in the IA scenario, the dominant factor is the REQ packet duration and clock phases difference instead of carrier sensing. Also, comparing Figs. 9 and 10, we observe that in both guard-time and no-guard-time systems, clock phases play a similar role as the REQ packet duration. The crossover point in Fig. 10 is when θ_1 offsets the REQ packet duration, i.e., $\theta_1 = -R$.

V. Modeling Arbitrary Topologies

Due to spatial reuse in a multihop network, several flows can transmit simultaneously during a cycle. Thus, our Markovian model can be naturally extended to the multihop case using one state for each such noninterfering set of flows. Then, the success probability of each flow can be found by adding the stationary probabilities of the noninterfering flow sets it belongs. However, we do not pursue this approach due to: 1) complexity of enumerating the independent sets and computing transition probabilities under imperfect synchronization, and 2) accurate prediction of per-flow success probability requires global information.

Instead, we introduce a model that computes a *lower bound* on the success probability of S-CSMA systems with guard time, based only on one-hop information. This approach not only simplifies analysis, but also enables to directly study the factors that affect fairness properties and ultimately enable distributed algorithms that address unfairness by providing minimum throughput guarantees.

The key observation that enables development of such a model in a S-CSMA system with guard time is the fact that, at the beginning of each cycle, all flows: 1) reset their contention windows to minimum; 2) always sense an idle channel due to the use of guard time that exceeds the maximum clock drift

in the network; and 3) they compute their backoff counters independently. This allows finding a lower bound on the success probability of each flow by expressing the one-hop neighborhood interference of each flow in product form. This interference expression is a lower bound to the success probability of a flow because, in reality, its one-hop interferers can *themselves* experience interference by the two-hop neighbors of this flow. The second critical step that reduces the model complexity and leads to a closed-form expression is a classification of one-hop interferers according to their topological (dis)advantages.

A. Modeling Success Probability Lower Bound

We use a simple geometric interference model characterized by the transmission range R_{T} and sensing range R_{S}. R_{T} is defined as the maximum distance that allows correct decoding of a packet. R_{S} $(R_{\mathrm{S}} \geq R_{\mathrm{T}})$ is defined as the maximum distance that triggers carrier sensing as well as the maximum distance that can cause collision due to simultaneous reception of more than one transmission. We define the one-hop interfering set L_i of flow i as the set of flows whose transmitter or receiver is within sensing range of either the transmitter or receiver of flow i.

We now consider a flow i in isolation and derive a lower bound b_i on its success probability π_i as a function of the flows in L_i. We divide the one-hop interfering set L_i of flow i in three subsets.

F_i: *Set of Equivalent Neighbor Flows of Flow i:* Any flow j in this set would have equal success probability with flow i had they been contending in isolation using perfectly synchronized S-CSMA and equal contention windows. This set includes any flow j in L_i that has: 1) a common transmitter or receiver node with flow i; 2) a transmitter within sensing range of the flow i transmitter; 3) a transmitter not in sensing range of the flow i transmitter but a receiver within sensing range of the flow i receiver; or 4) a transmitter only within range of the flow i receiver and a receiver only within range of the flow i transmitter. We note that set F is symmetric, i.e., for any flow $j \in F_i$, it also holds that $i \in F_j$.

A_i: *Set of Advantaged Neighbor Flows of Flow i:* Any flow j in this set would have higher success probability than flow i due to the REQ packet duration had they been contending in isolation using perfectly synchronized S-CSMA and equal contention windows. Formally, this set includes all flows j in L_i for which neither transmitter nor receiver are within sensing range of the transmitter of flow i and whose transmitter is within range of the receiver of flow i.

D_i: *Set of Disadvantaged Neighbor Flows of Flow i:* For any flow j in this set, flow i is in A_j.

To compute b_i we assume flow i only contends with its one-hop neighbors. This is a pessimistic assumption for the success probability of flow i because its one-hop neighbors can themselves experience interference from the two-hop neighbors of flow i (namely flows in L_j that are not in L_i). This can only increase the success probability of flow i.

At the beginning of each cycle, the backoff counters computed by different flows are independent from each other. Let the backoff counter of flow i at the beginning of a cycle be x_i mini-slots. With respect to the set F_i, flow i will win the contention only if all the equivalent flows' backoff counters are greater than x_i.

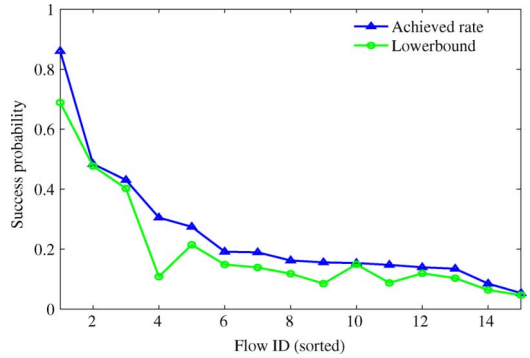


Fig. 11. Lower-bound evaluation in a random 15-flow multihop topology deployed in an area of 1000×1000 m^2. The lower bound is always lower than the achieved throughput and follows the same trend as the throughput.

TABLE II
MAC-LAYER PARAMETERS

SIFS	10 μs
DIFS	50 μs
σ	20 μs
BasicRate	2 Mbps
DataRate	2 Mbps
PLCP length	192 bits @ 1 Mbps
MAC header (REQ,GNT,ACK,DATA)	(20,14,14,28) bytes @ BasicRate
Packet size	1000 bytes
$(\mathrm{CW_{min}}, \mathrm{CW_{max}})$	(31,1023)
Retry Limit (Short,Long)	(7,4)
S-CSMA cycle duration	30ms
(Transmission Range, Sensing Range)	(250 m, 250 m)

With respect to A_i, flow i will win contention only if all the advantaged flows' backoff counters are greater than x_i plus REQ packet duration. Finally, with respect to D_i, flow i will win contention only if all the disadvantaged flows' backoff counters plus REQ packet duration are greater than x_i. Removing the condition on $X_i = x_i$ and incorporating relative phases, we reach the following expression for b_i:

$$b_i = \sum_{x_i=0}^{W_i-1} p_i(x_i) \prod_{f \in F_i} \Phi_f(x_i + \theta_{if}) \times \prod_{a \in A_i} \Phi_a(x_i + R + \theta_{ia}) \times \prod_{d \in D_i} \Phi_d(x_i - R + \theta_{id}). \quad (8)$$

Derived under a pessimistic assumption, b_i is a lower bound to the success probability π_i of each flow i. To investigate how b_i is related to π_i, we performed extensive ns simulations using the MAC-layer parameters specified in Table II. We consider 50 15-flow multihop random topologies deployed in an area of 1000×1000 m^2. Each flow has a clock drift that is randomly selected from a uniform distribution of 0–31 mini-slots. We next measure the achieved throughput for each flow and calculate the corresponding lower bound. Fig. 11 depicts the results of a representative example, which plots π_i and b_i for each flow i, sorted in decreasing order of π_i. The lower bound curve, b_i, is below the throughput curve, π_i, and follows its decreasing trend.

In order to confirm that the results are not specific to the representative example, we also calculate the average ratio between b_i and π_i (i.e., b_i/π_i) and the corresponding standard deviation across all the flows and over all the simulation topologies. The average value of b_i/π_i and the corresponding standard deviation are equal to 0.84 and 0.12, respectively. In addition, b_i is

always lower than π_i. Thus, (8) indeed presents a close lower bound on the throughput and presents a good approximation of network congestion.

B. Impact of REQ Packet Duration

Here, we derive an approximate closed form of (8) to investigate the impact of REQ packet size on the success probability lower bound. Starting from (8), we first set all phase terms to zero to isolate the effect of REQ packet duration. We then consider an approximate backoff model where each backoff counter X_i is geometrically distributed with parameter $q_i(= 2/W_i)$ instead of being uniformly distributed within $[0, W_i - 1]$. This approximation has also been successfully used in asynchronous CSMA protocol models [17], [27]. In the geometric backoff model, q_i is the probability that flow i transmits in the next mini-slot if its current window is W_i. Next, we further approximate (8) by its continuous form

$$b_i = \int_{x_i=0}^{\infty} p_i^c(x_i) \prod_{f\in F_i} \Phi_f^c(x_i) \times \prod_{a\in A_i} \Phi_a^c(x_i + R) \times \prod_{d\in D_i} \Phi_d^c(x_i - R)dx_i \quad (9)$$

where $p_i^c()$, $\Phi_i^c()$ are the continuous counterparts of $p_i()$,$\Phi_i()$, respectively.

Since X_i in discrete form is geometrically distributed with parameter $q_i = 2/W_i$, in continuous form it is exponentially distributed with parameter $\lambda_i = 2/W_i$. Substituting $p_i^c()$ and $\Phi_i^c()$ corresponding to this distribution in (9) and after algebraic manipulations, we reach the following closed-form expression for the lower bound of flow i:

$$b_i = \frac{\lambda_i e^{-R(C_a - C_d)}}{\lambda_i + C_f + C_a + C_d} \quad (10)$$

where $C_f = \sum_{f\in F_i} \lambda_f$, $C_a = \sum_{a\in A_i} \lambda_a$, and $C_d = \sum_{d\in D_i} \lambda_d$. According to (10), b_i is jointly determined by the one-hop interferers of flow i in the set L_i, yet each subset F_i, A_i, and D_i of L_i contributes differently. Next, we investigate the contribution of each individual subset.

Impact of Equivalent Interfering Flows F_i: The expression for b_i considering only the flows in F_i is obtained by setting $D_i = \emptyset$, $A_i = \emptyset$ in (10)

$$b_i = \frac{\lambda_i}{C_f + \lambda_i} = \frac{1}{\frac{W_i|F_i|}{\widetilde{W}_f} + 1}. \quad (11)$$

Here, $|F_i|$ is the number of flows in F_i and $\widetilde{W}_f$ is the harmonic mean of the contention windows of the flows in F_i.[5]

Eq. (11) reveals that when flow i contends only with equivalent flows, b_i does not depend on the REQ packet duration. Eq. (11) also shows that the window ratio $W_i/\widetilde{W}_f$ can be used to control the throughput ratio between flow i and its interfering flows. To illustrate application to fairness concepts, we use a simple scenario where all interferers of flow i are *not* within range of each other. The interference relationships can be represented by a star-shaped flow contention graph centered at flow i (Fig. 12), where vertices correspond to flows and edges correspond to pairwise interference among flows. If all flows are backlogged, the sum of their success probabilities within each clique of the flow contention graph should equal one.[6] Also, in this scenario, the success probability of each flow equals its lower bound. Therefore, $b_f = 1 - b_i$ for every interfering flow f of flow i.

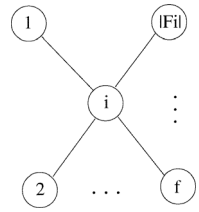


Fig. 12. Flow contention graph of flow i and $|F_i|$ independent interferers.

It is straightforward to show that a max-min fair allocation in the scenario of Fig. 12 would be $b_i = b_f = 1/2$. This can be achieved by $W_i/\widetilde{W}_f = 1/|F_i|$ in (11). On the other hand, under a proportionally fair allocation, flow i should receive $b_i = 1/(|F_i| + 1)$ and each interferer f should receive $b_f = |F_i|/(|F_i| + 1)$. This can be achieved by $W_i/\widetilde{W}_f = 1$ in (11).

Impact of Advantaged Interfering Flows A_i: The expression for b_i considering only the flows in A_i is obtained by setting $F_i = \emptyset$, $D_i = \emptyset$ in (10)

$$b_i = \frac{\lambda_i e^{-RC_a}}{C_a + \lambda_i} = \frac{e^{-\frac{2R|A_i|}{\widetilde{W}_a}}}{\frac{W_i|A_i|}{\widetilde{W}_a} + 1} \quad (12)$$

where $|A_i|$ is the number of flows in A_i and $\widetilde{W}_a$ is the harmonic mean of the contention windows of the flows in A_i. Similar to the case of set F_i, (12) shows that b_i depends on the ratio $W_i/\widetilde{W}_a$. However, b_i decreases exponentially with REQ packet duration R. The disadvantage due to the REQ packet duration cannot be addressed by flow i decreasing its contention window W_i (the exponential term persists even if W_i is zero). It can only be addressed by increasing $\widetilde{W}_a$.

We now rearrange terms in (12) and express W_i as a function of $\widetilde{W}_a$, R, and b_i

$$W_i = \frac{\widetilde{W}_a}{|A_i|b_i}\left(e^{-\frac{2R|A_i|}{\widetilde{W}_a}} - b_i\right). \quad (13)$$

Eq. (13) gives the contention window pairs $(W_i, \widetilde{W}_a)$ that achieve allocation b_i subject to REQ packet duration R and number of interferers $|A_i|$.

We experimentally validate (13) in our testbed, using a three-flow topology with flow i and two advantaged interfering flows ($|A_i| = 2$). We perform five experiments. In each experiment, a window pair $(W_1, \widetilde{W}_a)$ is set to values predicted by (13) to achieve max-min fair b_i using $R = 3.2$ mini-slots (REQ size in our implementation). We then measure the success probabilities π_i and compute Jain's Fairness Index ($f = (\sum_{i=1}^{i=n} p_i)^2/n \times \sum_{i=1}^{i=n}(p_i)^2$). The data points on the $R = 3.2$ line in Fig. 13 show that, in all cases, the contention

[5] $\widetilde{W}_f = (\sum_{f\in F_i} |F_i|^{-1} W_f^{-1})^{-1}$.

[6] This holds because success probabilities reflect the number of flow transmission opportunities. In contrast, the sum of normalized throughputs would be less than one due to contention overhead.

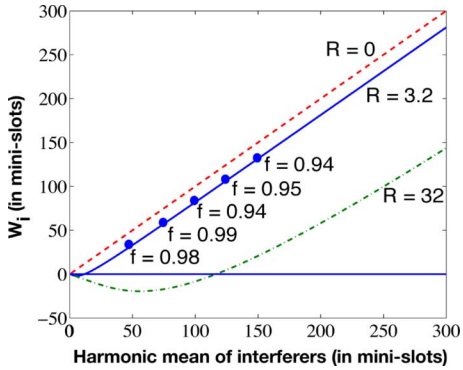


Fig. 13. Contention windows yielding max-min fair transmission for various values of REQ packet size. Jain's fairness index (f) of experimental results is calculated for the circled data points.

windows predicted by the model achieve throughput allocation very close to a max-min fair allocation ($f = 1.0$).

Fig. 13 also plots (13) for various REQ packet sizes. For each curve, data points above the $W_i = 0$ line correspond to a feasible pair $(W_i, \widetilde{W}_a)$ that yield max-min fairness. For larger R, the required contention window sizes to achieve max-min fairness are higher. Larger contention windows require a larger cycle to absorb the overhead due to the contention phase. However, a larger cycle yields higher delays in the network.

Impact of Disadvantaged Interfering Flows D_i: The expression for b_i considering only the flows in D_i is obtained by setting $F_i = \emptyset$, $A_i = \emptyset$ in (10), i.e.,

$$b_i = \frac{\lambda_i e^{RC_d}}{C_d + \lambda_i} = \frac{e^{\frac{2R|D_i|}{\widetilde{W}_d}}}{\frac{W_i|D_i|}{\widetilde{W}_d} + 1} \tag{14}$$

where $|D_i|$ is the number of flows in D_i and $\widetilde{W}_d$ is the harmonic mean of the contention windows of the flows in D_i. Eq. (14) indicates that the throughput of flow i increases rapidly to one as the size of REQ packet increases.

C. Impact of Clock Phase Differences

We now investigate the impact of clock phase differences to the success probability lower bound b_i predicted by our model. When the relative phases θ_{if}, θ_{ia}, and θ_{id} are nonzero, the closed-form approximation of (8) can be derived using the same procedure as in Section V-B. Here, we provide the closed-form expression for some special cases of interest. More specifically, we consider the worst case where flow i is lagging all its interfering flows (all phase terms θ_{if}, θ_{ia}, and θ_{id} are nonnegative) and investigate the isolated effect due to each interference set.

Impact of Equivalent Interfering Flows F_i: In this case, we are interested in how fast the throughput of flow i decreases when its clock lags behind the flows that contend fairly with flow i. We consider the case where F_i are the only contending flows of flow i and the clock of flow i lags behind all flows in F_i, i.e., $\theta_{if} \geq 0$. With these conditions, (8) is approximated by

$$b_i = \frac{\lambda_i e^{-\sum_{f \in F_i} \lambda_f \theta_f}}{C_f + \lambda_i}. \tag{15}$$

We further consider two special scenarios. In the first scenario, all flows in F_i have the same contention window size W, resulting in $\lambda_f = \lambda = 2/W$. Let M and μ respectively denote the number of flows in F_i and the mean of their relative clock phase with respect to the clock of flow i. Eq. (8) is then simplified as

$$b_i = \frac{\lambda_i e^{-\lambda M \mu}}{C_f + \lambda_i}. \tag{16}$$

An interesting observation is that a single-hop network is a special case of the scenario we consider here if we think flow i is the slowest flow in the single-hop network and other flows in F_i are other flows in the same network. In this case, the lower bound closely matches the success probability, when the collision probability is small.

In the second scenario, we let all flows in F_i have the same clock phase θ relative to the clock of flow i, i.e., $\theta = \theta_i - \theta_j$. Eq. (15) is then written as

$$b_i = \frac{\lambda_i e^{-C_f \theta}}{C_f + \lambda_i} \tag{17}$$

which predicts that the lower bound of flow i's success probability decreases exponentially with θ.

Impact of Advantaged Interfering Flows A_i: The closed-form expression for b_i when only advantaged flows are considered reads as follows:

$$b_i = \frac{e^{-2|A_i|\left(\frac{R}{\widetilde{W}_a} + \frac{\widehat{\Theta}_{ia}}{\widetilde{W}_a(\phi_{ia})}\right)}}{\frac{W_i|A_i|}{\widetilde{W}_a} + 1} \tag{18}$$

where $\widetilde{W}_a(\phi_{ia})$ is the harmonic mean weighted by the normalized relative phasesϕ_{ia},[7] and $\widehat{\Theta}_{ia}$ is the average relative phase with respect to flow i.[8]

Upon comparison to the perfectly synchronized case [(12)], we observe that the average relative phase adds to the REQ packet duration disadvantage in the exponential term of (18). In addition, this combined disadvantage in the exponential term can only be addressed by taking into account the contribution of the relative clock phase of each interferer. This can be done by increasing the $\widetilde{W}_a(\phi_{ia})$ and $\widetilde{W}_a$ values as captured in (18).

Impact of Disadvantaged Interfering Flows D_i: This is the interesting case where flow i is advantaged with respect to all interferers, yet it lags in phase with respect to all of them. The final expression for b_i in this case is given by

$$b_i = \frac{e^{-2|D_i|\left(\frac{\widehat{\Theta}_{id}}{\widetilde{W}_d(\phi_{id})} - \frac{R}{\widetilde{W}_d}\right)}}{\frac{W_i|D_i|}{\widetilde{W}_d} + 1} \tag{19}$$

where all quantities are defined similar to (18). The main observation here is that when the REQ packet duration R is equal to $\widehat{\Theta}_{id}\widetilde{W}_d/\widetilde{W}_d(\phi_{id})$, the impact of REQ packet size is canceled by the relative clock phases and thus the exponential term becomes unity.[9] At this point (19) becomes (11)—flow i

[7] $\widetilde{W}_a(\phi_{ia}) = (\sum_{a \in A_i} \phi_{ia} W_a^{-1})^{-1}$, where $\phi_{ia} = \theta_{ia} / \sum_{a \in A_i} \theta_{ia}$.

[8] $\widehat{\Theta}_{ia} = \sum_{a \in A_i} \theta_{ia} / |A_i|$.

[9] A special case is the crossover point in Fig. 10(b) that depicts the behavior of the guard-time system in the IA scenario.

competes with its disadvantaged interfering flows as if they were all equivalent in a perfectly synchronized system. However, as $\widehat{\Theta}_{id}\widetilde{W}_d/\widetilde{W}_d(\phi_{id})$ increases beyond R, b_i decreases exponentially with the throughput captured through (19).

VI. Model Application: Addressing Unfairness via Contention Window Adjustment

In this section, we discuss the application of our model to design simple contention window adjustment mechanisms aimed at addressing unfairness in S-CSMA. Our proposed solution aims at addressing unfairness at the link (MAC) layer. Designing end-to-end fairness mechanisms for multihop sessions requires information exchange between the link layer and transport layer, which is an interesting avenue for future research.

A. Fairness Objective

The closed-form expression for b_i provided by (10) can be potentially used in an optimization-based approach to achieve a fairness objective such as proportional fairness or max-min fairness. For example, a proportional fairness formulation would try to find contention windows W_i (input variables) that maximize the quantity $\sum_i \log(b_i)$ subject to (10). This approach has been successfully used in ALOHA [28] or high-signal-to-noise-ratio (SNR) CDMA [29] systems, where the throughput expressions allow for convex transformations. Such transformations are not possible in our system because the input variables in (10) appear in both fractions and exponents. Still, (10) has the interesting property that b_i is expressed as a function of only four input terms: W_i and the harmonic means $\widetilde{W}_f$, $\widetilde{W}_a$, $\widetilde{W}_d$ of the interfering groups of this flow. This property could be exploited to make the problem more tractable. In addition to complexity, our analysis in Section V demonstrates that if REQ packet size, average clock drift, or number of interferers are large, max-min fairness or proportional fairness in synchronized CSMA may require very large contention window sizes even for simple scenarios such as Fig. 12.

Due to the above reasons, we leave an optimization-based approach for future work. Here, we propose a new alternative approach with a much simpler objective: Consider a reference rate for each flow in the network and design a contention window adjustment algorithm that exceeds the minimum reference rate for that flow. The challenge here is to identify a "good" set of reference rates. These reference rates should have the following three properties.

- *Property 1:* Allow computation using only local information—contention window adjustment algorithm must be distributed.
- *Property 2:* Be sufficiently high (certainly nonzero) and reflect congestion in the network—flows with more interferers should have a lower reference rate than flows with less interferers.
- *Property 3:* Be sufficiently low, so that they can be exceeded by the contention window adjustment mechanism.

We believe that this is an interesting idea for addressing unfairness or starvation not only for synchronized CSMA, but also for other types of access protocols or networks. For the case of S-CSMA, we have found the following set of reference rates/success probabilities:

$$r_i = \frac{1/\left(|L_i|+1\right)}{\prod_{j\in L_i}\left(1+\frac{1}{|L_j|}\right)} \tag{20}$$

where $|L_i|$ is the number of flows that interfere with flow i. The reference rates satisfy Property 1: To compute r_i, flow i only needs to know the number of its one-hop interfering flows as well as the number of their interfering flows. This information can be passed by each flow periodically by broadcasting the number of its one-hop interferers and piggybacking this information in REQ/GNT handshake [30]. Property 2 is also satisfied since the reference rate is inversely proportional to the number of one-hop and two-hop interfering flows. Property 3 is satisfied if we can find conditions where each flow under S-CSMA achieves higher success probability than its reference rate.

We now show that with appropriate selection of contention window sizes and under certain conditions, each flow can achieve a throughput that is higher than its reference rate. These conditions are: 1) contention window size is sufficiently large compared to the size of REQ packet and maximum clock drift; 2) contention phase duration is sufficiently large compared to the contention window size. With these conditions, in the case of perfect synchronization, the term $e^{-R(C_a-C_d)}$ approaches 1. Eq. (10) then becomes

$$b_i = \frac{\lambda_i}{C_f + C_a + C_d + \lambda_i}. \tag{21}$$

From Eq. (21), if we assign the same contention window to all flows in the network, we obtain $b_i = 1/|L_i|$. Note, however, that according to (20), the reference rate for each flow is less than or equal to $1/(|L_i|+1)$. Thus, the lower bound for each flow, and therefore its throughput, would be higher than the reference rate.

B. Contention Window Adjustment (CWA) Algorithm

We now describe a simple distributed heuristic contention window adjustment algorithm, called CWA. All functions for each flow are implemented at its transmitter node. At each time t, each flow i maintains a throughput estimate $T_i^{(t)}$ and a reference rate estimate $r_i^{(t)}$. The reference rate estimate requires knowledge of two-hop neighborhood and its frequency of calculation is dependent on the mobility of the nodes as well as the traffic burstiness[10] of the flows.

Two-hop information is provided by each flow broadcasting at certain measurement instants the number of its interferers $|L_i|$. It also broadcasts a bit that characterizes itself as "satisfied" or "non-satisfied." The one-hop neighbor flows use this information to maintain their own estimates.

At certain sampling instants (equally or less frequent than the measurement instants), the flow compares its current throughput estimate to its current reference rate estimate. If the throughput estimate is *greater than or equal* to the reference rate estimate, the flow sets its satisfied bit to one and performs no contention window adjustment. Otherwise, it sets its satisfied bit to zero.

[10]We assume that traffic and mobility change at a slower timescale than the time required for reference rate calculation and window adjustment. Addressing unfairness under high mobility or rapidly changing traffic is an interesting topic for future work.

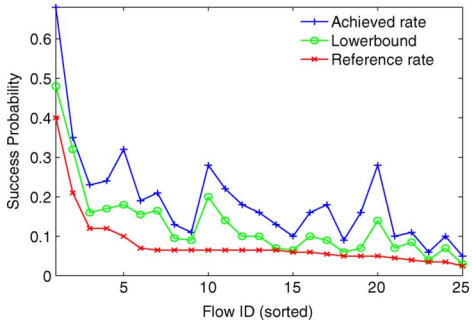


Fig. 14. 25 single-hop, perfectly synchronized flows in a multihop network.

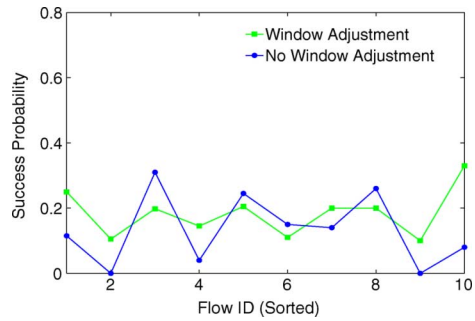


Fig. 15. 10 multihop, imperfectly synchronized flows in a grid network.

Next, the flow selects an arbitrary one-hop neighbor flow that is satisfied (if such a flow exists) and requests it to increase its contention window size. The satisfied flow receiving such a request increases its window by one mini-slot. This request step is necessary because according to the analysis in Section V, the unfairness due to phase and REQ packet size can only be addressed by increasing the contention windows of the interferers.

There are two choices for implementing throughput estimation. The first is for the flow to measure its success probability online by counting its successes over several cycles. The second is to use the lower bound approximation b_i from (10). The approximation b_i has the advantage to run the algorithm much faster (at packet level) without requiring measurements over several cycles. On the other hand, it requires each flow to maintain and broadcast some additional information: maintain its one-hop neighborhood contention window sizes and clock phases, broadcast its contention window size, and unicast to each one-hop neighbor its corresponding clock phase. We use the second approach in the experiments that follow.

For simplicity, our definition of the base rate in (20) and the proposed CWA algorithm do not incorporate data packet losses due to wireless channel conditions. Such losses can be incorporated by multiplying the base rate definition [(20)] and the calculated lower-bound approximation b_i of each flow i in the CWA algorithm with the corresponding data packet success probability.

C. Evaluation

In this section, we evaluate the performance of CWA using ns simulations. Unless otherwise specified, we use the PHY and MAC parameters of Table II in the ns simulations.

Experiment 1: Countering REQ Packet Size Impact: We first evaluate the ability of CWA to address the topological (dis)advantages caused only by the REQ packet size. We randomly place 25 one-hop flows with perfectly synchronized clocks in a 1000×1000 m^2 region. All flows have an initial contention window size equal to 64 mini-slots. Fig. 14 depicts the flow reference rates sorted in decreasing order and the corresponding lower bound estimates produced by CWA and the ns throughput results. We observe that both the lower bound b_i and the actual throughput curves are above the reference rates' curve. In addition, the lower bound curve is below the throughput curve and follows it closely, indicating that b_i is both a good congestion indicator and a good estimation of the actual throughput.

Experiment 2: Countering Clock Phase and REQ Packet Size Problems in Multihop Topologies: We now apply CWA to multihop flows with imperfect synchronization to counter both REQ packet size and clock drift. We are particularly interested in upload/download scenarios where there exist: 1) contention between upstream/downstream links of the same flow, and 2) contention between different flows that are in transmission range. Due to lack of space, we present the simulation results of a grid topology. Similar results were observed in other topologies with which we have experimented. We place 10 five-hop horizontal flows on a grid. Each outer flow interferes with one neighbor flow, and each intermediate flow interferes with only two neighbor flows.

Fig. 15, compares the per-flow throughput achieved before and after application of CWA. We observe that before CWA several flows starve, while after CWA their throughput significantly increases.

VII. Conclusion

In this paper, we analyzed the performance of synchronized CSMA contention under clock drifts. We showed that in single-hop systems, early flows achieve high throughput, whereas the throughput of the late flows decreases sharply with clock drift. In multihop systems where clock phase differences are coupled with carrier sense and topological bias, we showed that carrier sense in no-guard-time systems can act as a protection mechanism against clock drifts. On the other hand, guard-time systems offer more predictability of throughput. Finally, we applied the findings of our model to design a simple contention window adjustment mechanism that aims at addressing unfairness in S-CSMA systems. CWA is a first approach on addressing the inherent clock drifts and spatial bias in S-CSMA networks. We believe that joint design of CWA and clock synchronization mechanisms would provide further benefits and is an interesting avenue for future research.

References

[1] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications*, IEEE Std. 802.11, 1999.

[2] J. So and N. Vaidya, "Multi-channel MAC for ad hoc networks: Handling multi-channel hidden terminals using a single transceiver," in *Proc. ACM MobiHoc*, Tokyo, Japan, May 2004, pp. 222–233.

[3] R. Maheshwari, H. Gupta, and S. Das, "Multichannel MAC protocols for wireless networks," in *Proc. IEEE SECON*, Reston, VA, Sep. 2006, vol. 2, pp. 393–401.

[4] P. Bahl, R. Chandra, and J. Dunagan, "SSCH: Slotted seeded channel hopping for capacity improvement in IEEE 802.11 ad-hoc wireless networks," in *Proc. ACM MobiCom*, Sep. 2004, pp. 216–230.

[5] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, "Span: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks," in *Proc. ACM MobiCom*, Rome, Italy, Jul. 2001, pp. 85–96.

[6] W. Ye, J. Heidemann, and D. Estrin, "Medium access control with coordinated adaptive sleeping for wireless sensor networks," *IEEE/ACM Trans. Netw.*, vol. 12, no. 3, pp. 493–506, Jun. 2004.

[7] Y. Sun, S. Du, O. Gurewitz, and D. Johnson, "DW-MAC: A low latency, energy efficient demand-wakeup MAC protocol for wireless sensor networks," in *Proc. ACM MobiHoc*, Hong Kong, China, May 2008, pp. 53–62.
[8] M. Park, S. Choi, and S. Nettles, "Cross-layer MAC design for wireless networks using MIMO," in *Proc. IEEE GLOBECOM*, St. Louis, MO, 2005, vol. 5, pp. 2870–2874.
[9] N. Sato and T. Fujii, "A MAC protocol for multi-packet ad-hoc wireless network utilizing multi-antenna," in *Proc. IEEE CCNC*, Las Vegas, NV, Jan. 2009, pp. 1–5.
[10] K. Romer, "Time synchronization in ad hoc networks," in *Proc. ACM MobiHoc*, Long Beach, CA, Oct. 2001, pp. 173–182.
[11] A. Marco, R. Casas, J. Ramos, V. Coarasa, A. Asensio, and M. Obaidat, "Synchronization of multihop wireless sensor networks at the application layer," *IEEE Wireless Commun. Mag.*, vol. 18, no. 1, pp. 82–88, Feb. 2011.
[12] J. P. Sheu, C. M. Chao, W. K. Hu, and C. W. Sun, "A clock synchronization algorithm for multihop wireless as hoc networks," *Wireless Pers. Commun.*, vol. 43, no. 2, pp. 185–200, Oct. 2007.
[13] M. Carvalho and J. J. Garcia-Luna-Aceves, "A scalable model for channel access protocols in multihop ad hoc networks," in *Proc. ACM MobiCom*, Philadelphia, PA, Sep. 2004, pp. 330–344.
[14] M. Garetto, T. Salonidis, and E. Knightly, "Modeling per-flow throughput and capturing starvation in CSMA multi-hop wireless networks," *IEEE/ACM Trans. Netw.*, vol. 16, no. 4, pp. 864–877, Aug. 2008.
[15] A. Kashyap, S. Ganguly, and S. Das, "A measurement-based approach to modeling link capacity in 802.11-based wireless networks," in *Proc. ACM MobiCom*, Montreal, Canada, Oct. 2007, pp. 242–253.
[16] L. Qiu, Y. Zhang, F. Wang, M. Han, and R. Mahajan, "A general model of wireless interference," in *Proc. ACM MobiCom*, Montreal, Canada, Oct. 2007, pp. 171–182.
[17] G. Sharma, A. Ganesh, and P. Key, "Performance analysis of contention based medium access control protocols," in *Proc. IEEE INFOCOM*, Barcelona, Spain, Apr. 2006, pp. 1–12.
[18] M. Durvy and P. Thiran, "A packing approach to compare slotted and non-slotted medium access control," in *Proc. IEEE INFOCOM*, Barcelona, Spain, Apr. 2006, pp. 1–12.
[19] X. Wang and K. Kar, "Throughput modelling and fairness issues in CSMA/CA based ad-hoc networks," in *Proc. IEEE INFOCOM*, Miami, FL, USA, 2005, vol. 1, pp. 23–34.
[20] O. Yang and W. Heinzelman, "Modeling and throughput analysis for SMAC with a finite queue capacity," in *Proc. IEEE ISSNIP*, Melbourne, Australia, Dec. 2009, pp. 409–414.
[21] T. Zheng, S. Radhakrishnan, and V. Sarangan, "Modeling and performance analysis of DMAC for wireless sensor networks," in *Proc. ACM MSWiM*, Miami, FL, Oct. 2011, pp. 119–128.
[22] O. Yang and W. Heinzelman, "Modeling and throughput analysis for X-MAC with a finite queue capacity," in *Proc. IEEE GLOBECOM*, Miami, FL, Dec. 2010, pp. 1–5.
[23] K. Kar, S. Sarkar, and L. Tassiulas, "Achieving proportional fairness using local information in Aloha networks," *IEEE Trans. Autom. Control*, vol. 49, no. 10, pp. 1858–1863, Oct. 2004.
[24] M. Garetto, T. Salonidis, and E. Knightly, "Modeling per-flow throughput and capturing starvation in CSMA multi-hop wireless networks," in *Proc. IEEE INFOCOM*, Barcelona, Spain, 2006, pp. 1–13.
[25] Rice University, Houston, TX, "Rice University WARP project," [Online]. Available: http://warp.rice.edu
[26] C. Chaudet, I. Guerin-Lassous, and B. Gaujal, "Study of the impact of asymmetry and carrier sense mechanism in IEEE 802.11 multi-hops networks through a basic case," in *Proc. ACM PE-WASUN*, Venice, Italy, 2004, pp. 1–7.
[27] A. Kumar, E. Altman, D. Miorandi, and M. Goyal, "New insights from a fixed point analysis of single cell IEEE 802.11 WLANs," in *Proc. IEEE INFOCOM*, Miami, FL, Mar. 2005, vol. 3, pp. 1550–1561.
[28] X. Wang and K. Kar, "Cross-layer rate control in multi-hop wireless networks with random access," in *Proc. ACM MobiHoc*, Urbana-Champaign, IL, May 2005, pp. 157–168.
[29] M. Chiang, "Balancing transport and physical layers in wireless multihop networks: Jointly optimal congestion control and power control," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 1, pp. 104–116, Jan. 2005.
[30] X. Huang and B. Bensaou, "On max-min fairness and scheduling in wireless ad-hoc networks: Analytical framework and implementation," in *Proc. ACM MobiHoc*, Long Beach, CA, Oct. 2001, pp. 221–231.

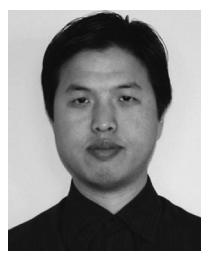
Ehsan Aryafar (S'05–M'11) received the B.S. degree in electrical engineering from Sharif University of Technology, Tehran, Iran, in 2005, and the M.S. and Ph.D. degrees in electrical and computer engineering from Rice University, Houston, TX, in 2007 and 2011, respectively.

He is a Post-Doctoral Research Associate with Princeton University, Princeton, NJ. His research interests are in the areas of wireless networks, high-performance MAC protocol design, network deployment and resource provisioning, and network measurements. His current projects include white space networking and hybrid satellite and terrestrial networks.

Theodoros Salonidis (S'98–M'04) received the Diploma in electronic and computer engineering from the Technical University of Crete, Chania, Greece, in 1997, and the M.S. and Ph.D. degrees in electrical engineering from the University of Maryland, College Park, in 1999 and 2004, respectively.

He is a Research Staff Member with the IBM T. J. Watson Research Center, Yorktown Heights, NY. He was a Post-Doctoral Researcher with Rice University, Houston, TX, from 2004 to 2006, and a Researcher with Intel Research, Cambridge, U.K in 2006 and Thomson/Technicolor, Paris, France, from 2007 to 2012. From 1999 to 2000, he was a Research Intern with the IBM T. J. Watson Research Center. His current research interests are in performance analysis, design, and implementation of mobile wireless networks and smart grid systems.

Dr. Salonidis is a member of the Association for Computing Machinery (ACM) and the Technical Chamber of Greece.



Jingpu Shi (S'96–M'07) received the B.S. degree from the University of Electronic Science and Technology of China, Chengdu, China, in 1996, the M.S. degree from The Ohio State University, Columbus, in 2004, and the Ph.D. degree from Rice University, Houston, TX, in 2007, all in electrical and computer engineering.

He is a Quantitative Analyst with Doliver Capital Advisors L.P., Houston, TX. From 1996 to 2001, he was a Researcher with the Institute of Semiconductors, Chinese Academy of Sciences, Beijing, China. His research interests are in the areas of wireless networks, high-performance MAC protocol design, and quantitative finance.

Edward Knightly (S'91–M'96–SM'04–F'09) received the B.S. degree from Auburn University, Auburn, AL, in 1991, and the M.S. and Ph.D. degrees from the University of California, Berkeley, in 1992 and 1996, respectively, all in electrical engineering.

He is a Professor of electrical and computer engineering with Rice University, Houston, TX. His research interests are in the areas of mobile and wireless networks and high-performance and denial-of-service resilient protocol design.

Prof. Knightly is a Sloan Fellow. He served as Associate Editor of numerous journals and special issues including the IEEE/ACM TRANSACTIONS ON NETWORKING and the IEEE JOURNAL ON SELECTED AREAS OF COMMUNICATIONS Special Issue on Multi-Hop Wireless Mesh Networks. He served as Technical Co-Chair of IEEE INFOCOM 2005, General Chair of ACM MobiHoc 2009 and ACM MobiSys 2007, and served on the program committee for numerous networking conferences including ICNP, INFOCOM, MobiCom, and SIGMETRICS. He is a recipient of National Science Foundation CAREER Award. He received the Best Paper Award from ACM MobiCom 2008.