This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

MAC-layer Time Fairness across Multiple Wireless LANs

Ming Zhang Shigang Chen
Department of Computer & Information Science & Engineering
University of Florida
Gainesville, FL 32611, USA

Ying Jian
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052, USA

Abstract*—Wireless LANs have been densely deployed in many urban areas. Contention among nearby WLANs is location-sensitive, which makes some hosts much more capable than others to obtain the channel for their transmissions. Another reality is that wireless hosts use different transmission rates to communicate with the access points due to attenuation of their signals. We show that location-sensitive contention aggravates the throughput anomaly caused by different transmission rates. It can cause throughput degradation and host starvation. This paper studies the intriguing interaction between location-sensitive contention and time fairness across contending WLANs. Achieving time fairness across multiple WLANs is a very difficult problem because the hosts may perceive very different channel conditions and they may not be able to communicate and coordinate their operations due to the disparity between the interference range and the transmission range. In this paper, we design a MAC-layer time fairness solution based on two novel techniques:** *channel occupancy adaptation, which applies AIMD on the channel occupancy of each flow, and** *queue spreading***, which ensures that all hosts and only those hosts in a saturated channel detect congestion and reduce their channel occupancies in response. We show that these two techniques together approximate the generic adaptation algorithm for proportional fairness.**

I. Introduction

Today, cities are crowded with wireless access points. For example, 12,276 access points were detected in London according to a RSA report in October 2008 [1]. At airport terminals, office buildings, homes and shops, a laptop can often find several to a few dozens of access points in usable range, most of which run the IEEE 802.11b/g protocol and each supports a WLAN. So many WLANs are located near one another. With only three non-overlapping channels in 802.11b/g, communications in different WLANs will inevitably interfere with each other, giving rise to two intriguing problems.

The first problem is location-sensitive contention [2]. The contention resolution protocol, 802.11 DCF (Distributed Coordination Function), works well in a symmetric environment where all hosts are downloading content via the same access point. But it does not perform well in asymmetric settings that are common when hosts in nearby WLANs contend in the same channel. Depending on their relative spatial locations, some hosts may gain huge advantage in occupying the channel for their transmissions. As they obtain most of the channel bandwidth, hosts in other WLANs are starved.

Location-sensitive contention is very difficult to deal with because of two fundamental wireless properties. *The first property* is that contention is defined by the interference or carrier-sensing range, whereas communication happens within the transmission range. Consequently, contending hosts in different WLANs may not be able to explicitly exchange or implicitly overhear necessary information to coordinate their operations. They may not even know whom they contend with. *The second property* is that hosts in different WLANs may sense very different channel conditions (in terms of channel idle time, transmission failure rate, or buffer occupancy) even when they contend in the same channel. When they observe the same channel in different states, if they cannot communicate (due to the first property), their reactions are bound to be different, causing unfairness. An extensive discussion on this issue can be found in [3].

The second problem is time-allocation anomaly. IEEE 802.11b allows four different transmission rates, 11Mbps, 5.5Mbps, 2Mbps and 1Mbps. IEEE 802.11g or 802.11a allows eight different rates. The transmission rate of a wireless host is determined through auto rate fallback based on how reliably the host can communicate with the access point at a certain rate. The transmission rate will be lower if the host is further away from the access point or there are obstacles (such as walls) between them. It is well known that, if a single host in a WLAN chooses a low transmission rate, all other hosts in the same WLAN suffer with low throughput [4]. To address the above anomaly, researchers have proposed to replace throughput fairness with time fairness, in which all hosts occupy the channel for the same fraction of time. However, most prior work does not consider the impact of location-sensitive contention that exists among nearby WLANs.

On one hand, location-sensitive contention can push time-allocation anomaly to the extreme, allowing a low-rate host in one WLAN to obtain an excessive amount of channel time and starve the high-rate hosts in a nearby WLAN. On the other hand, location-sensitive contention makes time-allocation anomaly a much harder problem to solve because contending hosts in different WLANs may be outside of each other's transmission range (but within the interference or carrier-sensing range). Unlike a multihop wireless network that has a communication path between any two nodes, in our WLAN setting, there may not exist any node between two contending hosts to relay their information. Most prior solutions for time fairness have largely ignored the impact of location-sensitive contention. They either rely on a central coordinator [4], [5], [6], or assume that each host has certain knowledge about its contending hosts [7], that the contending hosts will sense the same channel condition [8], [9], or that all hosts in all WLANs mutually contend [3]. These assumptions do not hold in general.

In this paper, we solve the time fairness problem under location-sensitive contention among multiple WLANs. Loosely speaking, our solution, called AIMD/QS+k, ensures that each wireless host will receive a fair share of the channel time even when it has no way to know exactly whom it contends with. Precisely speaking, AIMD/QS+k approximately achieves proportional fairness in channel time allocation for the hosts across multiple WLANs. The design of AIMD/QS+k is based on two novel techniques, called *channel occupancy adaptation* and *queue spreading*. The former applies additive increase multiplicative decrease on each host's channel occupancy. The latter accurately identifies which hosts saturate the channel during the time of congestion. For example, suppose two hosts, x and y, in different WLANs cause congestion. Due to location advantage, x can send out all its packets at the expense of lowered throughput at y. Hence, only y detects congestion. To resolve congestion, both y and x should perform multiplicative decrease. If y reduces its channel occupancy alone, x will simply pick up the extra bandwidth and widen the unfairness. Without any means of communication, how can y make sure that x (but none of its other contending hosts that do not contribute to the current congestion) will join the channel-occupancy reduction? Remarkably, this problem can be solved in a fully distributed way without requiring the nodes to exchange information, overhear, or even know each other's existence. We demonstrate that AIMD/QS+k achieves almost perfect proportional fairness in scenarios where the existing solutions fail.

The rest of the paper is organized as follows. Section II gives the network model and problem definition. Section III describes the time-allocation anomaly and the location-sensitive contention. Section IV discusses the related work. Section V proposes our AIMD/QS+k solution. Section VI presents the simulation results. Section VII draws the conclusion.

II. Network Model and Problem Definition

A. Network Model

Consider a number of wireless access points that are deployed in an area. Each access point connects one or more wireless hosts to the Internet. Each host has a transceiver that can either transmit or receive at a time. An access point and its hosts form a WLAN. The access point selects a channel, i.e., a sub-band of the available frequency range, to communicate with its hosts. IEEE 802.11b/g has 11 channels, among which only three are non-overlapping. Transmissions in nearby WLANs that select the same or overlapping channels may interfere with one another. However, spatial channel reuse happens among distant WLANs that can use the same channel to transmit simultaneously without interference.

We use a *node* to refer to either an access point or a wireless host. The sequence of data packets sent from a node x to a node y constitute a *MAC flow* (x, y). The *transmission range* of node x defines the distance within which another node is able to decode the data transmitted by x. The *interference range* of node y defines the distance within which another node's transmission will interfere with y's reception of a packet from x. The *carrier-sensing range* of node x defines the distance within which another node's transmission will cause x to sense a busy channel. To avoid collision, it should be set no less than the maximum interference range, which can be 1.78 times the transmission range as suggested in [10] (also the default value used in ns-2).[1].

We assume a DCF-like MAC protocol. Two MAC flows contend if the following conditions are met: (1) their transmissions are made in the same channel or in channels with overlapping frequency bands that cause interference, and (2) the sender of a flow can carrier-sense the transmission of the other flow or the receiver of a flow is interfered by the transmission of the other flow. If RTS/CTS is turned on, then contention also happens if the receiver of a flow can carrier-sense the transmission of the other flow. When the two flows (x, y) and (u, v) contend, we also say that node x has a contending node u.

The *transmission rate* of a MAC flow (x, y) is defined as the modulation rate of the sender x, at which x transmits its bits in the channel. The *channel occupancy* (or occupancy for short) of a flow is defined as the fraction of a unit time during which the flow occupies the channel for transmission. It includes both the time for transmitting data packets and the time for control packets such as ACKs. The *delivery rate* of (x, y) is defined as the average rate at which y can successfully receive data from x. It is bounded by the transmission rate multiplied by the channel occupancy.

A maximal set of mutually contending flows is called a *contention group*, also known as a contention clique in the prior work [12], [13]. The flows in a contention group have to take turn to transmit. The sum of the channel occupancies for all flows in a group is called the *aggregate occupancy of the group*, which is bounded by one. A contention group is said to be *saturated or congested* if no flow can further increase its delivery rate without decreasing the rate of another flow in the group. Two flows in different contention groups may be able to transmit simultaneously due to channel spatial reuse. An example is given in Fig. 1, which has two contention groups: g_1 consists of (w, z) and (x, y); g_2 consists of (x, y) and (u, v_i), $1 \le i \le 4$. Flows (w, z) and (u, v_i) can transmit at the same time.

B. Problem Definition

When there is only one contention group, the time fairness problem is to equalize the channel occupancies of all MAC flows while fully utilizing the channel capacity. However, when there are more than one contention group, it is not always possible to equalize the flows' channel occupancies because each flow experiences different contention. Proportional fairness [14] has been introduced for such cases.

Let F be the set of all MAC flows, G be the set of contention groups, and $q = (q_f, f \in F)$ be a vector of *feasible* channel occupancies, such that $q_f \ge 0$, $\forall f \in F$, and $\sum_{f \in g} q_f \le Q_g$, $\forall g \in G$, where Q_g is the maximum aggregate occupancy that g can have before it is congested. Q_g will be smaller than one due to the protocol overhead of DCF. A feasible vector q^* is *proportionally fair* if for any other feasible vector q, the sum of the proportional changes is non-negative [14], i.e., $\sum_{f \in F} \frac{q_f^* - q_f}{q_f^*} \ge 0$. It has been shown in [14] that this is

[1]Note that the carrier-sensing range can be *artificially configured*. It can be made to equal the transmission range [11]. This however does not alter the fact that contention goes beyond the transmission range because the interference range is not a quantity that can be artificially configured. Reducing the carrier-sensing range to be smaller than the interference range increases the chance of collision

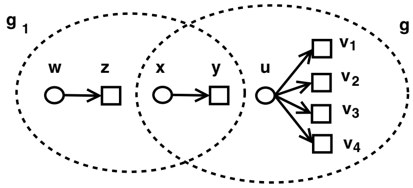


Fig. 1. Three WLANs form two contention groups g_1 and g_2.

equivalent to find a feasible vector q that maximizes the sum of a utility function $U(q_f) = \ln(q_f)$.

$$\begin{aligned} & maximize \sum_{f \in F} U(q_f) \\ & subject\ to \sum_{f \in g} q_f \leq Q_g, \forall g \in G \\ & \qquad q_f \geq 0, \forall f \in F. \end{aligned} \tag{1}$$

Exactly solving the problem in a run-time environment is extremely hard due to the complex interaction among the contending flows, as we will elaborate in this paper. Our goal is to approximate the proportional time-fairness through a fully distributed, DCF-compatible solution. DCF-compatibility means that the solution does not require any modification to the DCF protocol or its random backoff algorithm, but it may retrieve some state information from the MAC layer and modify some MAC parameters (such as the size of the minimum contention window) on the fly.

We stress that the concept of contention group is introduced only to help us describe our solution. The operations in our distributed solution *never* need to actually identify which flows belong to each contention group or know the value of Q_g. In fact, Q_g does not even have to be a constant. As it may drift over time, the optimal value for q_f will also evolve. Our solution will dynamically converge the channel occupancies of the flows towards their current optimal values. Unlike the previous work that constructs the contention groups explicitly [12], [13], our solution does not do so because this is not always feasible in our general setting where nodes that contend may not know each other's identities. For example, in Fig. 1, if the distance between z and x is greater than the transmission range but smaller than the interference range, then x has no way to know if it contends with one link or ten links.

III. Time-Allocation Anomaly and Location-Sensitive Contention

The time-allocation anomaly among hosts in a single WLAN is a well known fact [4]. DCF ensures that the MAC flows in a WLAN have equal chance to send their packets. In other words, each MAC flow will send roughly the same number of packets over long run. Suppose the flows have roughly the same average packet size. It will take more time for a flow with a smaller transmission rate to transmit a packet. Hence, DCF ends up giving more channel time to a MAC flow with a smaller transmission rate, which effectively keeps the channel operate at the smaller rate more often, reducing the WLAN's overall throughput.

We observe that time-allocation anomaly exists among hosts in nearby WLANs even when the transmission rates of all hosts

Fig. 2. Two MAC flows in different WLANs contend.

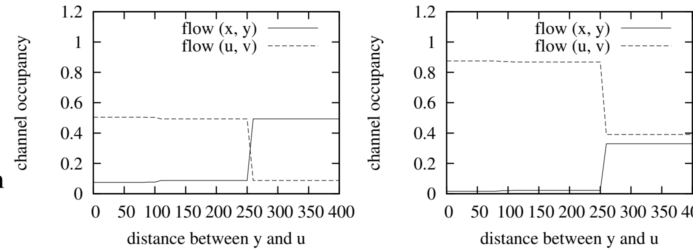


Fig. 3. Channel occupanies of the flows in Fig. 2 under DCF. Left plot: both flows transmit at 11 Mbps. Right plot: flow (x, y) transmits at 11 Mbps and flow (u, v) at 1 Mbps. Simulation setup in ns2: The length of each wireless link is 150m, the interference range is 1.78 times the length of the wireless link [10], the transmission range is 250m, and the carrier-sensing range is 550m. The packet size is 1000 bytes.

are the same. We perform simulation on the network in Fig. 2, which models the scenario of two neighboring homes whose WLANs operate in the same channel. All simulations in the paper are performed on ns-2 [15]. When the transmission rates of both links are 11 Mbps, the simulation result in the left plot of Fig. 3 demonstrates that the channel occupancies of the two flows are very different and dependent on the distance between y and u. The hidden-terminal problem arising in Fig. 2 was first documented and analyzed in [16], which, however, does not consider the fact that the carrier-sensing range and interference range are greater than the transmission range. Readers are referred to [3] for a much more detailed explanation on the cause of location-sensitive contention in this network.

Now, if we add different transmission rates on top of location-sensitive contention, they drive the time-allocation anomaly problem to the extreme. In Fig. 2, when the transmission rate of (x, y) is 11 Mbps but the transmission rate of (u, v) is 1 Mbps, the simulation result in the right plot of Fig. 3 shows that the channel occupancy of (x, y) is almost zero when the distance between y and u is less than 250 m.

IV. State of the Art

In this section, we discuss the distributed time fairness solutions without a central coordinator [4], [5], [6]. TFCSMA [7] assumes that each node knows the *number of its contending nodes*, which is needed in computing the node's target throughput (equivalent to the delivery rate in this paper). This is however not always feasible. Suppose the carrier-sensing range is twice of the transmission range. Because the area within the carrier-sensing range is four times the area within the transmission range, the number of contending nodes that cannot be identified can be much greater than the number of nodes that can be identified, which will seriously affect the performance of TFCSMA. A similar problem exists for all solutions that require nodes to collect each other's information through overhearing.

Idle Sense [8] assumes that all nodes sense the same idle time in the channel. It works well for a single WLAN, but not for multiple WLANs. For example, in Fig. 2, suppose node u is within the interference range of y but outside the carrier-sensing range of x. Even when the channel is saturated by

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

flow	rate	occupancy	flow	rate	occupancy
(w, z)	123.26	0.161	(u, v_2)	123.27	0.161
(x, y)	112.35	0.147	(u, v_3)	123.16	0.161
(u, v_1)	123.17	0.161	(u, v_4)	123.23	0.161

TABLE I
DELIVERY RATE (IN PACKETS PER SECOND) AND CHANNEL OCCUPANCY UNDER PISD ON THE NETWORK OF FIG. 1

the transmission on (u, v), node x will sense an idle channel! Even if all nodes sense the same idle time, Idle Sense may still work poorly for multiple WLANs due to location-sensitive contention, which will be demonstrated in Section VI. The work by Nandagopal et al. [9] assumes that the senders of all flows in a contention group will have the same transmission failure probability, which is also not true among contending hosts in different WLANs [3].

Packets are fragmented in [17]. Flows with smaller transmission rates will have smaller fragment sizes. If the MAC flows transmit for roughly the same number of times and each time transmit one fragment, then their channel occupancies can be equalized. The opposite solution is to aggregate the packets [18], [19]. Flows with larger transmission rates will have larger packet aggregates. Both solutions assume that DCF gives all MAC flows equal chance to transmit. This is true in the symmetric setting when the flows belong to the same WLAN. However, it is not true in asymmetric settings when the flows belong to different WLANs, which is evident from the simulation result in the left plot of Fig. 3.

CWSP [20] makes the size of the minimum contention window inversely proportional to the transmission rate. It decreases the probability for a flow with a small transmission rate to obtain the channel and consequently reduces its channel occupancy. However, this heuristic approach cannot bring quantitative precision and does not always work well (Section VI).

PISD [3] applies proportional increase synchronized multiplicative decrease to control the rate of each flow. Its main objective is to achieve throughput fairness although it may be extended to approximate time fairness. However, it makes an assumption that all flows belong to a single contention group. When the contention group is saturated, any node that detects the congestion will jam the channel to help other nodes also detect the congestion. This ensures that the nodes will perform multiplicative decrease simultaneously. However, because *all* nodes will participate in jamming, for a large deployment of WLANs that form many partially-overlapping contention groups, jamming will spill out to other contention groups that are not saturated. It leads to *cascaded jamming*. It begins from one node at a congested hot spot. Its jamming causes the nearby nodes to detect congestion because they can hardly send out packets. When these nodes start their jamming, the nodes further away will feel the impact. As the process repeats, nodes outside of the congested contention group will falsely detect congestion and unnecessarily reduce their rates.

The above observation is confirmed by the simulation on the network in Fig. 1, which has two overlapping contention groups: g_1 has two mutually contending flows and g_2 has five flows. The two groups share a common flow (x, y). One would expect the delivery rate of (w, z) will be high because channel spatial reuse occurs between (w, z) and (u, v_i), $1 \leq i \leq 4$, since they do not contend. Let the transmission rates of all flows be 11 Mbps. The simulation result is shown in Table I. Due to cascade jamming, the delivery rate of (w, z) is comparable to the rates of the flows that belong to the bottleneck g_2. The total occupancy of g_2 is high. The total occupancy of g_1 is however very low. The reason is that, when g_2 is congested and x performs jamming, not only u in g_2 will feel it, but w in g_1 will also feel it.

Tremendous progress has been made to apply the theory of utility optimization in wireless networks. However, the existing work assumes different network models and thus cannot be directly applied to solve our problem. Some existing solutions require a centralized, NP-hard scheduling algorithm for all wireless links [21], [22]. Others assume a time-slotted cellular network model [23] or assume a node-exclusive interference model [24], [25], [26] where links can transmit simultaneously as long as they do not share a common node.

There is a large body of work on rate fairness in multihop wireless networks (MWN). It may appear that the WLAN networks studied in this paper are special cases of the general MWNs. This is not true. They are different problems and the WLAN networks may pose greater challenges. For one, neighbors in an MWN can exchange information (such as the case in [2], [27]), whereas the contending hosts in nearby WLANs may be too far to communicate and the WLANs can be viewed as components of a partitioned network.

V. A SOLUTION FOR MAC-LAYER TIME FAIRNESS ACROSS MULTIPLE WLANS

In this section, we propose our time fairness solution for WLANs, whose design is based on two novel techniques, called channel occupancy adaptation and queue spreading, which together approximately achieve proportional fairness among hosts in contending WLANs.

A. Overview

The basic idea of our solution is to apply the AIMD (additive increase multiplicative decrease) adaptation on channel occupancy in each contention group. More specifically, all flows in a contention group perform AIMD on their channel occupancies. When the aggregate occupancy of the group does not cause channel congestion, each flow in the group will increase its channel occupancy by a constant amount, which is *additive increase*. When the aggregate occupancy of the group becomes too large and causes channel congestion, all flows in the group will decrease their channel occupancies by a certain percentage, which is *multiplicative decrease*.

If there is only one contention group, then AIMD will indeed equalize the channel occupancies of all flows. The gap between the largest occupancy l and the smallest occupancy s among all flows stays the same after each additive increase but shrinks after each multiplicative decrease. That is because, as both l and s are reduced by the same percentage, l is reduced by a larger absolute amount, and hence their gap shrinks. That gap diminishes over time after multiplicative decrease is performed periodically.

When there are multiple contention groups and each node may participate in more than one group, then AIMD — which is performed independently in each group — may not equalize the channel occupancies of all flows. As we will discuss in

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

Section V-E, it approximately achieves proportional fairness. For example, in Fig. 1, flow (x, y) participates in two contention groups. In proportional fairness, the channel occupancy of (x, y) will be modestly smaller than those of other flows in g_2 because (x, y) also participates in multiplicative decrease in g_1 (which happens less frequently because g_1 only has two flows and it takes longer time for additive increase to congest the channel). This is generally regarded as a positive feature because it strikes a balance between fairness and throughput improvement. A smaller (but not too small) channel occupancy for (x, y) improves the channel spatial reuse between (w, z) and (u, v_i), $1 \leq i \leq 4$, as they can transmit simultaneously.

Our idea for time fairness faces two major technical problems. The first problem is how to actually perform AIMD on channel occupancy. The second problem is most challenging: When a contention group is congested, we want *all* flows in the group and *only* the flows in the group to perform multiplicative decrease. Consider the example in Fig. 1. Suppose g_2 is congested. Among all flows in g_2, (x, y) resides at the most disadvantageous location. As (x, y) is least capable of obtaining the channel, it will be the first to detect the channel congestion. If x performs multiplicative decrease to resolve the congestion, other flows in g_2 will continue performing additive increase to pick up the bandwidth it has given up. Hence, node x must perform certain operations that cause u (but not w) to also detect the congestion. Recall that the nodes, x, u and w, may not be able to directly communicate or overhear each other. The rest of the section will solve these problems.

B. Release Rate and Channel Occupancy Adaptation

The control function of the proposed solution resides at the network layer. It does not require any modification to the DCF protocol, but must be able to query for the queue length at the MAC layer and modify the size of the minimum contention window. It adapts the channel occupancy of a flow by controlling the *release rate*, at which the network layer releases packets into the MAC layer.

When the combined release rate of all flows in a contention group is small, the channel can forward all packets released to the MAC layer. Hence, the delivery rate is equal to the release rate, and the MAC-layer queue remains empty. The flows can improve their channel occupancies by releasing more packets into the MAC layer. When the combined release rate of the flows in the group becomes too high and the aggregate channel occupancy is too large such that the channel is saturated, not all packets released to the MAC layer can be forwarded. Excess packets have to be buffered at the nodes' MAC-layer queues. Eventually, one node will observe that its queue length persistently grows and exceeds a threshold. The node will perform the operation of queue spreading (in the next subsection), which ensures that the queue lengths at all other nodes in the congested contention group will also exceed the threshold (such that they all detect the congestion). Then, the nodes should reduce their channel occupancies by decreasing the release rates, and fewer packets are made available to the MAC layer in order to relieve the congestion.

Intuitively, the queue-length threshold for congestion detection should be proportional to the transmission rate of the flow. That is because, under time fairness, a flow with a low transmission rate will have a small release rate, which would make its queue harder to pass the threshold (thus harder to detect congestion) if the threshold was a constant. Hence, we define the threshold as $H \times r$, where H is a system wide constant and r is the flow's transmission rate.

The *protocol for channel occupancy adaptation* is given as follows: Consider an arbitrary MAC flow (x, y) with a transmission rate r. The sender x adapts its channel occupancy after each time period of T. If its queue length at the MAC layer is below the threshold $H \times r$, it will additively increase its channel occupancy by a constant α. If the queue length reaches the threshold, it will multiplicatively decrease its channel occupancy by a percentage β. To implement multiplicative decrease on the channel occupancy, x simply reduces its release rate by a percentage β. To implement additive increase on the channel occupancy, x increases its release rate by $\alpha \times r$ (because it takes α time to transmit $\alpha \times r$ bits at rate r). Hence, while all flows share the same parameter α, the increments for their release rates, $\alpha \times r$, will be proportional to their transmission rates.

C. Queue Spreading

We propose a new technique, called *queue spreading*, which makes sure that the senders of all flows in a congested contention group will detect the congestion and they will perform multiplicative decrease together.

The aggregate release rate $R_g(t)$ of all flows in a contention group g is a function of time t. $R_g(t) = \sum_{f \in g} R_f(t)$, where $R_f(t)$ is the release rate of flow f. Let $C(t)$ be the maximum throughput that the group can possibly obtain from the channel at time t. We stress that $C(t)$ is only needed to describe our idea. The operation of queue spreading does not rely on the knowledge of $C(t)$. When $R_g(t)$ exceeds $C(t)$, if we look at the flows as a whole, there are more packets released to the MAC layer than it can send out. The excess packets increase the queue lengths of the flows at a combined rate of $R_g(t) - C(t)$. The problem is that, due to location-sensitive contention, most excess packets may be queued up at one flow that is least capable of accessing the medium. While that flow observes its queue length exceeds the threshold and performs multiplicative decrease, other flows more capable of obtaining medium may still find their queues empty and thus continue with additive increase, which will enlarge the gap among the flow rates and result in worse unfairness.

Our solution to the above problem is to *spread* the excess packets among the *queues* of all flows in the group. For an arbitrary flow (x, y) whose transmission rate is r, whenever the packet queue at the sender x exceeds $H \times r$, x will temporarily modify its MAC parameters to increase its ability of obtaining the medium, such that its queue length can be reduced back to $H \times r$. When the queue length becomes $H \times r$, the node will restore the original MAC parameters. The idea behind queue spreading is very intuitive: After a node detects congestion, the node will keep its queue length at $H \times r$ by dynamically adjusting its MAC parameters. Because its queue no longer grows, the excess packets in the channel will have to be buffered elsewhere, pushing the queues at other nodes up. Once their queues reach the threshold, they will do the same thing. Excess packets will always be pushed to the nodes that have not detected congestion yet.

A node that performs channel jamming [3] tries to *occupy the channel as much as possible*, and consequently it will affect all neighboring nodes, including those outside of the saturated group. On the contrary, a node that performs queue spreading only tries to *match its sending rate with its release rate* such that the local queue does not grow further. Therefore, it will not affect the neighbors outside of the saturated group. Let r_f be the transmission rate of f. We have the following proposition.

Proposition 1: The senders of all flows in a contention group g will detect congestion by the end of a time period $[t_0, t_1)$ if the following conditions are satisfied: (1) $R_g(t) - C(t) > 0,\ \forall t \in [t_0, t_1)$, (2) $\int_{t_0}^{t_1}(R_g(t) - C(t))dt \geq \sum_{f\in g} H \times r_f$, and (3) queue spreading is performed.

Proof: To prove by contradiction, we assume that a subset of flows, $g' \subset g$, does not detect congestion by time t_1. On one hand, the flows' packet queues are shorter than their respective thresholds. Thus, the total number of packets in their queues is less than $\sum_{f\in g'} H \times r_f$. On the other hand, by performing the operation of queue spreading, the flows in $g - g'$ are more capable of obtaining the medium than those in g'. Hence, they can control their queue lengths to the threshold values at the expense of the flows in g', whose queues will be growing. The total number of packets queued at the flows in $g - g'$ is $\sum_{f\in g-g'} H \times r_f$. During the time period $[t_0, t_1)$, by Condition (1), the total number of excess packets that are queued by all flows must be $\int_{t_0}^{t_1}(R_g(t) - C(t))dt$. Therefore, the number of packets queued at the flows in g' must be $\int_{t_0}^{t_1}(R_g(t) - C(t))dt - \sum_{f\in g-g'} H \times r_f$. By Condition (2), this number is no less than $\sum_{f\in g'} H \times r_f$, leading to the contradiction. □

It is easy to see that, if the congestion is detected by the senders of all flows in a group, then the total number of excess packets in their queues that the channel cannot deliver is at least $\sum_{f\in g} H \times r_f$. We have the following necessary condition for congestion detection.

Proposition 2: Suppose the senders of all flows in a contention group g have empty queues at time t_0 and $R_g(t) - C(t) > 0,\ \forall t \in [t_0, t_1)$. If all flows detect the congestion of g by time t_1, then we must have $\int_{t_0}^{t_1}(R_g(t) - C(t))dt \geq \sum_{f\in g} H \times r_f$.

D. AIMD/QS+k

We need to integrate queue spreading (QS) into the channel-occupancy adaptation protocol in Section V-B. However, a straightforward combination of the two will not do the trick. The first integrated protocol, called AIMD/QS, is given as follows: Each flow (x, y) performs the AIMD channel occupancy adaptation periodically as described in Section V-B. In addition, when the queue length at the sender x reaches the threshold, x performs queue spreading until the end of the current period T when it does multiplicative decrease. During the operation of queue spreading, if the queue length is above the threshold, the sender x aggressively reduces its minimum contention window to a small fraction of the default size in order to ensure that it has the priority to occupy the channel. Once the queue length is reduced to the threshold, x restores the default minimum contention window. By doing so, it keeps the queue length at the threshold.

Proposition 2 states that, in order for the senders of all flows in a congested group g to detect congestion, the number of excess packets buffered by all flows must be at least $\sum_{f\in g} H \times r_f$. However, AIMD/QS has no means to guarantee that.

To solve this problem, we design a generalized protocol, AIMD/QS+k, where k is a non-negative integer. The sender of each flow carries out the operations of AIMD/QS except that, after it finds its queue length reaches $H \times r$, it will continue performing additive increase for k subsequent periods of T before making multiplicative decrease. Suppose a flow's queue reaches $H \times r$ during $[t_0, t_0 + T)$. It will increase the release rate at times $t_0 + T$, $t_0 + 2T$, ..., $t_0 + kT$ by an amount $\alpha \times r$, and then decrease the release rate at time $t_0 + (k+1)T$ by a percentage β. The idea is to make sure that there will be enough excess packets to allow all nodes to detect congestion. The following proposition gives the formula for picking the parameters of AIMD/QS+k, such that when a contention group is congested, all flows in the group will detect the congestion and thus perform multiplicative decrease. (The flows outside of any congested group will not do that because there are no excess packets to push their queues over the threshold.)

Proposition 3: AIMD/QS+k ensures the detection of congestion by all flows in a congested group if

$$H \leq \frac{k(k-1)}{2}\alpha T, \text{ for } k \geq 2. \tag{2}$$

Proof: Consider an arbitrary time $t = 0$. Let g be the first contention group that becomes congested after $t = 0$. When g becomes congested, its constituent flows release more packets than the channel can deliver. The excess packets will eventually push the queue length of a flow (u, v) in g over the threshold. Without losing generality, suppose it happens during $(iT, (i+1)T]$. Let C be the channel capacity that can be maximally obtained by the group g at the moment. Clearly, $R_g((i+1)T) > C$. Otherwise, there would be no excess packets to push the queue length of (u, v) to the threshold.

Flow (u, v) will be the first one to perform multiplicative decrease and that will happen at time $t = (i+k+1)T$. Hence, $R_g(.)$ is a non-decreasing function before $t = (i+k+1)T$. Because the release rate of each flow f in g increases by $\alpha \times r_f$ after each period T, we must have $R_g((i+1+j)T) = R_g((i+1)T) + j\sum_{f\in g} r_f \alpha$. Hence,

$$\begin{aligned}
&\int_{(i+1)T}^{(i+k+1)T}(R_g(t) - C)dt \geq \sum_{j=0}^{k-1}(R_g((i+1+j)T) - C)T \\
&= \sum_{j=0}^{k-1}(R_g((i+1)T) - C)T + \sum_{j=0}^{k-1} j \sum_{f\in g} r_f \alpha T \\
&> \sum_{j=0}^{k-1} j \sum_{f\in g} r_f \alpha T \;=\; \frac{k(k-1)}{2}\sum_{f\in g} r_f \alpha T.
\end{aligned}$$

If (2) is met, we will have $\int_{(i+1)T}^{(i+k+1)T}(R_g(t) - C)dt > \sum_{f\in g} r_f \cdot H$. Hence, by Proposition 1, AIMD/QS+k makes sure that all flows in g detect the congestion. □

E. Proportional Fairness and Interpretation of AIMD/QS+k

In their seminal paper [14], Kelly, Maulloo and Tan show that there exist fully distributed algorithms that achieve global optimization of the system utility. Below we rewrite their primal algorithm in our notations (with simplification of removing the weight). Let F be the set of all flows and G_f be the set of contention groups to which flow f belongs.

$$\frac{d}{dt}q_f(t) = \alpha - q_f(t) \times \frac{1}{\varepsilon^2} \sum_{g \in G_f} (\sum_{f' \in g} q_{f'}(t) - Q_g + \varepsilon)^+. \quad (3)$$

The price functions $(\sum_{f' \in g} q_{f'}(t) - Q_g + \varepsilon)^+$ for $g \in G_f$ take the form suggested in [14], such that when $\varepsilon \to 0$, the above adaptation, when performed independently by the senders of all flows, will maximize the system utility, $\sum_{f \in F} \ln q_f$.

AIMD/QS+k can be interpreted as a discrete approximation of (3). The first item on the right side of (3) suggests additive increase. AIMD/QS+k increases q_f by a constant amount α after each time period of T. The second item on the right side suggests multiplicative decrease. $\sum_{g \in G_f} (\sum_{f' \in g} q_{f'}(t) - Q_g + \varepsilon)^+$ is the penalty factor. AIMD/QS+k decreases q_f by a percentage β when the accumulated penalty $\int_{t_0}^{t_1} \sum_{g \in G_f} (\sum_{f' \in g} q_{f'}(t) - Q_g + \varepsilon)^+ dt$ reaches a threshold, where t_0 is the time when the previous multiplicative decrease is performed and t_1 is the current time.

The key is to measure the penalty for each contention group g, $\int_{t_0}^{t} (\sum_{f' \in g} q_{f'}(t) - Q_g + \varepsilon)^+ dt$, which becomes positive only when g is congested. It is likely that, for any flow f, only one group in G_f is congested at a time if α is chosen small such that a contention group will spend much more time without congestion than time with congestion.

Interestingly, the price can be indirectly measured through the local queue length. Let $R_f(t)$ be the release rate of flow f and $r_f(t)$ be the transmission rate. In AIMD/QS+k, the channel occupancy q_f is controlled through the release rate R_f. In (3), q_f represents the *resource demand* of flow f. It is the channel occupancy that the flow demands in order to transmit all released packets. Hence, q_f is proportional to R_f (because each bit takes the same amount of time to transmit). After g is congested, further additive increase made by AIMD/QS+k will linearly increase the release rate (or the channel occupancy) of each flow. In the mean time, the number of excess packets that have to be buffered will increase proportionally. This implies that, approximately, the combined queue length for all flows in g grows at a speed proportional to the amount of excess channel occupancy, $(\sum_{f' \in g} q_{f'}(t) - Q_g)^+$. Furthermore, the technique of queue spreading ensures that the senders of all flows will see the same price as the excess packets are spread among the queues to push all of them over the threshold. Here, the price takes a discrete form, 0 if the threshold is not reached and 1 if the threshold is reached.

VI. Simulation

We perform extensive simulations on various scenarios to evaluate the performance of AIMD/QS+k in achieving MAC-layer time fairness. We compare AIMD/QS+k with some existing work, including 802.11 DCF, CWSP [20], and Idle Sense [8].

All simulations are performed on ns-2. AIMD/QS+k works on top of DCF. The parameters of DCF use the default values set by ns-2 according to the protocol standards. For the channel occupancy adaptation, $\alpha = 0.03$, $\beta = 0.5$, and $T = 1$ second. The parameters for queue spreading are determined based on Proposition 3. In particular, we choose $k = 2$ and $H = 0.03$ seconds, which satisfy (2) in the proposition. The parameters of other solutions are chosen based on the original papers.

We use three simulation scenarios with increasing complexity. The results from a simple scenario are easier to interpret, while the results from a complex scenario are closer to what will be seen in practice.

A. One Contention Group

The first simulation scenario is based on the network of Fig. 2, which contains only one contention group of two MAC flows, (x, y) and (u, v). The length of each wireless link is 150m, the interference range is 1.78 times the length of the wireless link [10], the transmission range is 250m, and the carrier-sensing range is 550m. The packet size is 1000 bytes. These parameters will also be used in other scenarios.

Fig. 4 compares the channel occupancies of the flows achieved under DCF, CWSP, Idle Sense, and AIMD/QS+k, respectively, with the transmission rate of (x, y) being 11 Mbps and the rate of (u, v) being 1 Mbps. Under DCF, the channel occupancy of (u, v) is much higher than the occupancy of (x, y). Under CWSP, the situation is opposite. The occupancy of (x, y) is better. Idle Sense performs very well until the distance is beyond 250m, where (u, v) is totally starved.

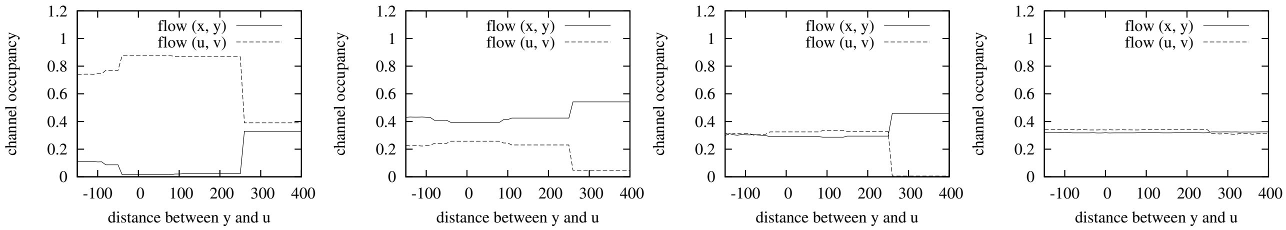
In real deployment of WLANs, RTS/CTS is often turned off. Our simulations find that the network throughput is consistently higher without RTS/CTS. The reason is that, according to the 802.11 standard, RTS/CTS are sent at the lowest transmission rate. Hence, they will take 656 μs at 1 Mbps, which represent significant overhead, considering that it only takes 940 μs to transmit a data packet of 1,000 bytes at 11 Mbps. We turn off RTS/CTS and re-run the simulation. The result is shown in Fig. 5. Indeed, the aggregate channel occupancy is improved in each case. AIMD/QS+k can maintain time fairness. The performance of Idle Sense improves when the distance is greater than 250m, but degrades when the distance is smaller than 250m as the 11-Mbps flow (x, y) is depressed by the 1-Mbps flow (u, v).

B. Two Contention Groups

The second simulation scenario uses the network of Fig. 1, which has two contention groups. Group g_1 contains two flows, (x, y) and (w, z). Group g_2 contains five flows, (x, y), (u, v_1), (u, v_2), (u, v_3), and (u, v_4). The length of each wireless link is 150m. The distance between z and x is 200m. The distance between y and u is also 200m. The transmission rate of (w, z) is 2 Mbps. The transmission rate of (u, v_3) is 1 Mbps. The transmission rates of other flows are 11 Mbps.

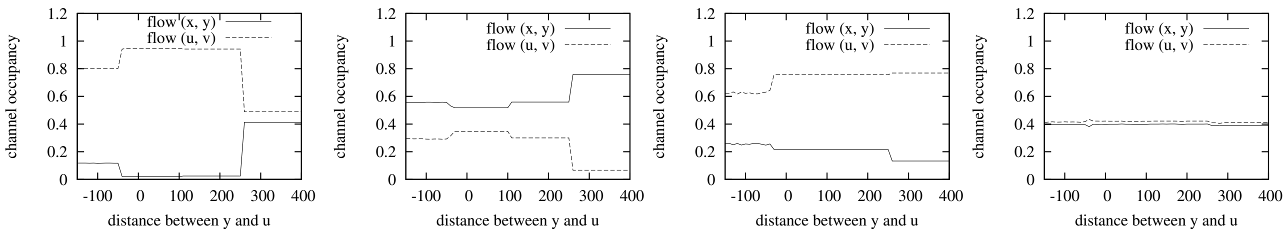
The simulation result is shown in Table II. Under DCF, (x, y) is starved and the occupancies of (u, v_1), (u, v_2) and (u, v_4) are very low due to the reason explained in Section III. Under either CWSP or Idle Sense, (x, y) is starved. Under AIMD/QS+k, (x, y) has a decent channel occupancy, even though it is smaller than others due to the nature of proportional fairness. Comparing with the result of PISD in Table I, AIMD/QS+k improves the channel occupancy of (w, z) from 0.161 to 0.546. We shall not compare the delivery rates because the transmission rate of (w, z) is 11 Mbps there and 2 Mbps here. The total throughput

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.



(a) 802.11 DCF (b) CWSP (c) Idle Sense (d) AIMD/QS+k

Fig. 4. Comparing the channel occupancies of flows (x, y) and (u, v) in the network of Fig. 2 under (a) 802.11 DCF, (b) CWSP, (c) Idle Sense, and (d) AIMD/QS+k, respectively. The transmission rates of (x, y) and (u, v) are 11 Mbps and 1 Mbps, respectively.



(a) 802.11 DCF (b) CWSP (c) Idle Sense (d) AIMD/QS+k

Fig. 5. Same as the caption of Fig. 4, but this time RTS/CTS is turned off.

of g_1 under AIMD/QS+k is comparable to the throughput under DCF or Idle Sense, but much better than the throughput under CWSP. The total throughput of g_2 under AIMD/QS+k is much better than the throughput under DCF or Idle Sense, but smaller than the throughput under CWSP.

C. Multiple Contention Groups

The third simulation scenario is designed based on the network of Fig. 6, where WLANs are deployed along two crossing streets. The relative positions of the nodes are drawn in the figure. The length of each wireless link is 150m. The distance between the closest nodes in two adjacent WLANs is 200m. The contention relationship among the flows is automatically determined in ns-2 based these parameters and those in Section VI-A. The transmission rates of some flows are specified in the figure, and the rates of others are 11 Mbps by default.

The simulation result is shown in Table III. Under DCF, flows 5, 11 and 15 have very low channel occupancies. A few others have low occupancies. Under CWSP, flows 8 and 14 have very low channel occupancies. Under Idle Sense, flows 1, 2, 3, 4, and 9 have very low channel occupancies. This is not a surprising outcome because Idle Sense was designed to work among hosts in a single WLAN. Under AIMD/QS+k, all flows have reasonable channel occupancies. Its overall distribution of channel occupancies is much fairer than those of others. We believe this simulation result demonstrates the strong performance of AIMD/QS+k under a complex scenario.

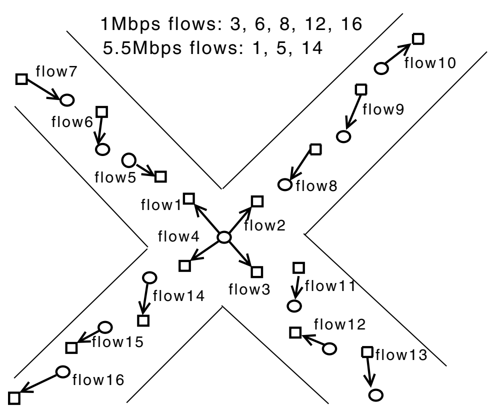


Fig. 6. WLANs are deployed along two streets. Unless specified in the figure, the default transmission rate of a flow is 11 Mbps.

VII. Conclusion

This paper proposes a new time fairness solution that approximates the generic adaptation algorithm for proportional fairness among multiple WLANs. The new solution addresses the problem of location-sensitive contention, and considerably outperforms the existing solutions. It is fully distributed. Each node only performs localized operations. It is DCF-compatible and only needs to modify the size of the minimum contention window.

VIII. Acknowledgments

This work was supported in part by the US National Science Foundation under grant CNS-0644033 and grant CPS-0931969.

References

[1] RSA Press Release, "Annual RSA Wireless Security Survey: With Encrypted Wi-Fi Vulnerable, How Companies and Individuals are Risking their Assets and Reputations," *http://www.rsa.com/press_release.aspx?id=9725*, October 2008.
[2] K. Xu, M. Gerla, L. Qi, and Y. Shu, "Enhancing TCP Fairness in Ad Hoc Wireless Networks using Neighborhood RED," In *Proc. of ACM MobiCom*, September 2003.
[3] Y. Jian and S. Chen, "Can CSMA/CA Networks Be Made Fair?," In *Proc. of ACM MobiCom*, September 2008.
[4] M. Heusse, F. Rousseau, G. Berger-Sabbatel, and A. Duda, "Performance Anomaly of 802.11b," In *Proc. of IEEE INFOCOM*, April 2003.
[5] G. Tan and J. Guttag, "Time-based Fairness Improves Performance in Multi-rate Wireless LANs," In *Proc. of USENIX Annual Technical Conference*, June 2004.

TABLE II
COMPARING THE DELIVERY RATES (IN PACKETS/SEC) AND THE CHANNEL OCCUPANCIES OF THE FLOWS IN THE NETWORK OF FIG. 1 UNDER 802.11 DCF, CWSP, IDLE SENSE, AND AIMD/QS+k.

	802.11 DCF		CWSP		Idle Sense		AIMD/QS+k	
flow	rate	occupancy	rate	occupancy	rate	occupancy	rate	occupancy
(w, z) at 2 Mbps	201.08	0.865	155.64	0.670	205.83	0.886	126.83	0.546
(x, y) at 11 Mbps	0.347	0.001	5.96	0.008	2.07	0.003	75.84	0.099
(u, v_1) at 11 Mbps	66.18	0.086	183.47	0.239	107.87	0.141	138.01	0.180
(u, v_2) at 11 Mbps	64.48	0.084	182.53	0.238	112.69	0.147	138.59	0.181
(u, v_3) at 1 Mbps	67.67	0.594	14.63	0.128	42.29	0.371	22.62	0.199
(u, v_4) at 11 Mbps	67.85	0.088	183.67	0.240	109.37	0.143	138.63	0.181

TABLE III
COMPARING THE DELIVERY RATES (IN PACKETS/SEC) AND THE CHANNEL OCCUPANCIES OF THE FLOWS IN THE NETWORK OF FIG. 6 UNDER 802.11 DCF, CWSP, AND AIMD/QS+k.

	802.11 DCF		CWSP		Idle Sense		AIMD/QS+k	
flow	rate	occupancy	rate	occupancy	rate	occupancy	rate	occupancy
flow1	52.88	0.108	79.79	0.164	3.29	0.007	61.12	0.125
flow2	51.18	0.067	169.37	0.221	4.54	0.006	81.74	0.107
flow3	54.45	0.452	13.23	0.116	0.193	0.002	20.59	0.181
flow4	52.27	0.068	139.29	0.182	6.25	0.008	77.97	0.102
flow5	21.41	0.044	48.55	0.099	303.07	0.622	78.03	0.160
flow6	97.18	0.853	36.03	0.316	6.85	0.060	42.51	0.373
flow7	37.29	0.076	226.47	0.464	367.19	0.753	169.03	0.347
flow8	15.11	0.133	1.92	0.017	61.01	0.536	17.31	0.152
flow9	73.08	0.095	72.82	0.095	23.01	0.031	169.95	0.222
flow10	469.73	0.613	538.69	0.702	132.73	0.173	257.31	0.336
flow11	10.35	0.013	76.78	0.100	120.99	0.158	86.21	0.112
flow12	89.71	0.788	7.51	0.066	45.31	0.398	43.30	0.380
flow13	107.07	0.140	549.74	0.717	349.47	0.456	264.33	0.345
flow14	31.41	0.064	1.87	0.004	195.93	0.402	43,48	0.089
flow15	16.82	0.022	362.29	0.472	63.43	0.083	203.11	0.265
flow16	105.77	0.929	40.15	0.352	95.09	0.835	60.01	0.527

[6] G. Tan and J. Guttag, "The 802.11 MAC Protocol Lead to Inefficient Equilibria," In *Proc. of IEEE INFOCOM*, March 2005.
[7] T. Joshi, A. Mukherjee, Y. Yoo, and D. Agrawal, "Air Time Fairness for IEEE 802.11 Multirate Networks," *IEEE Transactions on Mobile Computing*, vol. 7, no. 4, pp. 513–527, April 2008.
[8] M. Heusse, F. Rousseau, R. Guillier, and A. Duda, "Idle Sense: An Optimal Access Method for High Throughput and Fairness in Rate Diverse Wireless LANs," In *Proc. of ACM SIGCOMM*, August 2005.
[9] T. Nandagopal, T. Kim, X. Gao, and V. Bharghavan, "Achieving MAC Layer Fairness in Wireless Packet Networks," In *Proc. of ACM MobiCom*, August 2000.
[10] K. Xu, M. Gerla, and S. Bae, "How Effective is the IEEE 802.11 RTS/CTS Handshake in Ad Hoc Networks?," In *Proc. of IEEE GLOBECOM*, November 2002.
[11] J. Camp, V. Mancuso, O. Gurewitz, and E. W. Knightly, "A Measurement Study of Multiplicative Overhead Effects in Wireless Networks," In *Proc. of IEEE INFOCOM*, March 2008.
[12] X. Huang and B. Bensaou, "On Max-Min Fairness and Scheduling in Wireless Ad Hoc Networks: Analytical Framework and Implementation," In *Proc. of ACM MobiHoc*, October 2001.
[13] K. Jain, J. Padhye, V. N. Padmanabhan, and L. Qiu, "Impact of Interference on Multi-hop Wireless Network Performance," In *Proc. of ACM MobiCom*, September 2003.
[14] F. Kelly, A. Maulloo, and D. Tan, "Rate Control in Communication Networks: Shadow Prices, Proportional Fairness and Stability," *Journal of the Operational Research*, vol. 49, no. 3, pp. 237–252, 1998.
[15] "The Network Simulator - ns-2," *http://www.isi.edu/nsnam/ns/*.
[16] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: A Media Access Protocol for Wireless LANs," In *Proc. of ACM SIGCOMM*, August 1994.
[17] J. Dunn, M. Neufeld, A. Sheth, D. Grunwald, and J. Bennett, "A Practical Cross-Layer Mechanism for Fairness in 802.11 Networks," In *Proc. of International Conference on Broadband Networks (BROADNETS)*, October 2004.
[18] B. Sadeghi, V. Kanodia, A. Sabharwal, and E. Knightly, "Opportunistic Media Access for Multirate Ad Hoc Networks," In *Proc. of ACM MobiCom*, September 2002.
[19] T. Razafindralambo, I. Guerin-lassous, L. Iannone, and S. Fdida, "Dynamic and Distributed Packet Aggregation to Solve the Performance Anomaly in 802.11 Wireless Networks," *Computer Networks*, vol. 52, no. 1, pp. 77–95, January 2008.
[20] H. Kim, S. Yun, I. Kang, and S. Bahk, "Resolving 802.11 Performance Anomalies through QoS Differentiation," *IEEE Communications Letters*, vol. 9, no. 7, pp. 655–657, July 2005.
[21] A. Eryilmaz and R. Srikant, "Joint Congestion Control, Routing and MAC for Stability and Fairness in Wireless Networks," *IEEE Journal on Selected Areas in Communications, special issue on Nonlinear Optimization of Communication Systems*, vol. 24, no. 8, pp. 1514–1524, August 2006.
[22] M. Neely, E. Modiano, and C. Li, "Fairness and Optimal Stochastic Control for Heterogeneous Networks," *IEEE/ACM Transactions on Networking*, vol. 16, no. 2, pp. 396–409, April 2008.
[23] A. Eryilmaz and R. Srikant, "Fair Resource Allocation in Wireless Networks Using Queue-length based Scheduling and Congestion Control," In *Proc. of IEEE INFOCOM*, March 2005.
[24] X. Lin and N. B. Shroff, "The Impact of Imperfect Scheduling on Cross-Layer Congestion Control in Wireless Networks," *IEEE/ACM Transactions on Networking*, vol. 14, no. 2, pp. 302–315, April 2006.
[25] L. Chen, S. H. Low, M. Chiang, and J. C. Doyle, "Cross-layer Congestion Control, Routing, and Scheduling Design in Ad Hoc Wireless Networks," In *Proc. of IEEE INFOCOM*, April 2006.
[26] L. Bui, A. Eryilmaz, R. Srikant, and X. Wu, "Joint Asynchronous Congestion Control and Distributed Scheduling for Multi-hop Wireless Networks," In *Proc. of IEEE INFOCOM*, April 2006.
[27] S. Rangwala, A. Jindal, K. Y. Jang, K. Psounis, and R. GovindanL, "Understanding Congestion Control in Multi-hop Wireless Mesh Networks," In *Proc. of ACM MobiCom*, September 2008.