

Interaction Engineering: Achieving Perfect CSMA Handshakes in Wireless Networks

Vinay Kolar, IBM Research, India, vinkolar@in.ibm.com

Saqui Razak, Dept. of Computer Science, Carnegie Mellon University, Qatar, srazak@cmu.edu

Nael B. Abu-Ghazaleh, State University of New York, Binghamton, USA, nael@cs.binghamton.edu

Abstract—Carrier Sense Multiple Access (CSMA) protocols are unable to effectively arbitrate the medium in wireless networks; problems such as hidden and exposed terminals occur frequently leading to collisions, poor performance and unfairness. CSMA networks can be optimized by careful tuning of transceiver parameters, such as transmission power and carrier sensing threshold, to maximize spatial reuse of wireless channel while minimizing collisions. However, existing studies fail to jointly optimize these parameters to eliminate collisions and maximize spatial reuse. Our approach leverages on the observation that links under CSMA interfere in one of the few discrete interaction modes; each mode leads to different behavior in terms of performance and fairness. The proposed methodology controls the transceiver parameters to convert destructive interaction modes (such as various types of hidden terminals) into constructive ones; we call this approach *Interaction Engineering (IE)*. In this paper, we first formulate a model and centralized algorithm that computes the parameters based on one-to-one interaction between the links. We then develop a distributed IE protocol. We evaluate the protocols under Wireless LAN and multi-hop wireless networks using both simulation and testbed. We show that IE eliminates a vast majority of the collisions and significantly boosts spatial reuse. For example, in the WLAN scenarios, we observed a median improvement of $4\times$ in throughput and more than $2.5\times$ improvement in fairness, and orders of magnitude improvement in connection delay and jitter. IE also shows significant improvements in multi-hop networks, and under different forms of traffic such as video and TCP.

Index Terms—Wireless, Medium Access Control, CSMA, IEEE 802.11, Hidden terminals, Exposed terminals, Interactions

I. INTRODUCTION

Carrier Sense Multiple Access (CSMA) is a MAC layer algorithm widely used to coordinate access to the wireless medium in Wireless LAN and multi-hop networks. Under CSMA, a node transmits a packet only if the channel is sensed to be idle to reduce interference with other concurrent transmissions. Several popular protocols, such as IEEE 802.11 [1], use CSMA since it is well-suited for distributed implementation.

Senders using CSMA cannot arbitrate the wireless channel effectively, and may therefore experience poor performance and unfairness as they compete with other senders for the channel. If the sender senses the medium to be idle and transmits a packet when the receiver is experiencing interference, a hidden terminal occurs [2]. Similarly, if the sender unnecessarily defers transmission due to sensing the channel to be busy while the receiver's channel is idle, an exposed terminal occurs [3]. These effects can lead to suboptimal use of the channel, with poor performance and unfairness.

Wireless MAC protocols may be viewed to be solving a optimization problem with the following objectives: (i) maximize capacity by allowing concurrent transmissions when possible, and (ii) avoid collisions and other detrimental interactions between competing flows. The general approach we take to this problem is to *manipulate the interactions* between the transmitting links by controlling transceiver parameters to avoid harmful interactions. Specifically, each node may adjust parameters, such as transmission power and carrier sensing threshold to allow concurrent transmissions when they are possible and to avoid collisions when they are not. There is a complex inter-dependence between the settings chosen by each node. For example, a sender can increase transmission power to achieve higher Signal to Interference and Noise Ratio (SINR) for its link, but in the process create greater interference at other receivers. Similarly, the Carrier Sensing threshold determines the power level at which the channel is perceived as busy. Smaller values of this threshold reduce the possibility of hidden terminals, since the sender defers transmission even when there is a weak signal sensed on the channel. However, this high sensitivity may increase exposed terminals by needlessly preventing non-colliding concurrent transmissions. Since the solution must find effective settings for jointly for all active links, the problem is difficult.

This paper takes a new approach to optimizing the CSMA MAC problem that is based on insights CSMA behavior [4], [5]. These results demonstrate that interference is manifested through discrete interactions modes, rather than continuous metrics such as SINR. As a result, small changes in topology or radio parameters can significantly affect the performance by converting one destructive interaction mode into another more effective one. Thus, it is important to be aware of this behavior when optimizing transceiver parameters for a given network. Moreover, these studies identify that different interactions, beyond the simple hidden terminal/exposed terminal classification arise, with substantial impact on performance. Section II presents more information regarding the different interaction modes that arise in CSMA networks.

Although there are a number of existing studies that optimize CSMA behavior by manipulating transceiver parameters, our approach differs because it includes a wider range of possible MAC interactions that arise in a CSMA networks. In contrast, prior studies consider only a subset of topologies, parameters, or protocol rules. For example, many models assume a dense random topology [6], [7], tune only carrier sensing threshold [8], [9], [10], or assume that no ACKs are

sent [6], [7]. We discuss the related works in Section III.

An earlier version of this paper [11] presented the idea of Interaction Engineering (IE), where we jointly optimize the transmission power, carrier sense threshold, and receiver sensitivity to support concurrent transmissions while avoiding collisions. The paper presented an analytical model with a centralized solution, as well as a heuristic distributed solution. In this paper, we extend and generalize the model and associated centralized solution (Section IV) in a number of ways to allow optimization of wireless LANs and multi-hop networks. We develop a distributed protocol based on the new model in Section V. We extensively evaluate the protocol for both managed and unmanaged WLANs. We also evaluate the protocol for a number of new traffic scenarios.

In Section VI, we evaluate the effectiveness of IE in WLAN and multi-hop networks. We perform simulation and testbed experiments. In our WLAN simulation, we observe a median improvement of $4\times$ in throughput, $2.5\times$ in fairness and order of magnitude improvement in end-to-end delay and jitter. Even in the managed enterprise WLANs, we show an improvement of 50% in throughput, with significant improvements in fairness and delay. We also observe significant improvements for video and TCP traffic.

Experiments with a 6-node testbed demonstrate that IE is feasible in realistic deployments. We show that collisions are eliminated and spatial reuse is increased in links with strong SINR; without IE even such strong links were vulnerable to detrimental CSMA interactions.

The paper focuses on optimizing CSMA in a single channel with constant rate. Related studies consider additional dimensions such as channel assignment [12] and transmission rate-control [7], [13]. However, we believe that optimizing interactions among nodes that coexist in a single channel forms a basic block that recurs even while considering these additional parameters. We even show that Interaction Engineering boosts performance even with channel assignment algorithms used in existing Enterprise WLANs. In the future, we plan to extend this model to take advantage of these additional parameters to further optimize performance. Section VII overviews future work, and presents some concluding remarks.

II. BACKGROUND

In this section, we briefly overview the CSMA protocol. We then discuss the major interaction modes that arise in CSMA networks [4], [5].

A. CSMA protocol

In CSMA, the sender transmits a packet only when the channel is sensed to be idle. The Carrier Sensing Threshold (CS Threshold) parameter controls the signal level above which the channel is considered busy. The nodes monitor the channel and lock to any incoming signal if it is above the Receiver Sensitivity threshold (RS Threshold).

CSMA is susceptible to packet collision, where packet is lost due to interference from external sources. Due to the possibility of collisions and transmission errors in wireless

networks, protocols such as IEEE 802.11 use acknowledgments upon successfully receiving a DATA packet. When an ACK is not received, retransmit the packet; they use Binary Exponential Backoff (BEB) to reduce contention for the channel.

B. Interactions in CSMA networks

It is well known that CSMA networks suffer from hidden and exposed terminals [2], [3]. However, recent studies have shown that interference is more complex: two competing links can interact with each other in a number of different ways that exhibit different performance and fairness characteristics. These interactions can be categorized into a few discrete categories [4], [5] with different impact on throughput, delay and fairness. Figure 1 shows the commonly occurring interactions. *No Interaction (NI)*: Network performance is best when links within the network are interference free. Figure 1(a) shows two links that can be concurrently active since neither link interferes with the other link. We refer to this state of links to be *NI* (No Interference).

Sender Connected (SC): When the senders can sense each other's transmissions, they can better avoid collisions through carrier sense. The interaction between these two links is *Sender Connected (SC)* (Figure 1(b)). Here, both the links share the capacity equally without collisions (except when both senders transmit at exactly the same time, which is infrequent).

Hidden terminals: Traditionally, researchers have treated all forms of hidden terminals equally. Recent studies identified different forms of hidden terminals with varying throughput and fairness [4], [5]. Consider Figures 1 (c)(d)(e) that represent different forms of hidden terminals. In Figure 1(c), node S_2 is a hidden terminal for link S_1-D_1 . Packets at D_1 are dropped due to interference from S_2 , and S_1 experiences regular long backoff periods. S_2-D_2 dominates the channel at the expense of S_1-D_1 . This scenario called as *Asymmetric Incomplete State (AIS)* [4].

A similar interaction is *Symmetric Incomplete State (SIS)*, where both the senders are hidden terminal to other's receiver (Figure 1(c)). Neither link is able to transmit without collisions, leading to poor performance. Unlike AIS, SIS is fair in the long term as the interaction is symmetric.

In the third form of hidden terminals, the destinations cause collisions when they transmit their ACK packets. The throughput of this interaction is similar to SC but links suffer short term unfairness. This interaction, called *Interfering Destinations Incomplete State (IDIS)*, is depicted in Figure 1(e).

The fourth type of hidden terminal occurs due to Capture Effect [14]. The receiver D_1 locks to the signal from interferer S_2 . If the source S_1 transmits when D_1 has locked to S_2 , then D_1 ignores the S_1 's signal, causing a packet timeout. However, the packet is received successfully if D_1 locks to S_1 's transmission, and S_2 transmits at a later point in time. We refer to this interaction has *Hidden Terminal with Capture (HTC)* [14].

Other modes of interference can occur, but these have been shown through geometric analysis as well as experimentally to be rare [5]. Therefore, we consider only the categories explained above.

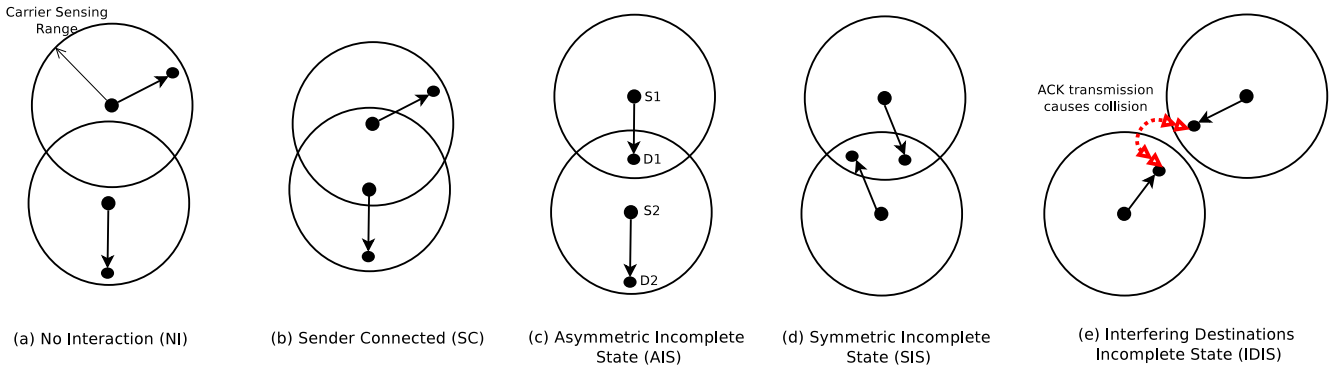


Fig. 1. Types of Interactions in CSMA.

III. MOTIVATION AND RELATED WORK

In this section, we motivate the CSMA optimization problem, and survey the related work. We then motivate the methodology of *Interaction Engineering* (IE) as a holistic approach for improving the performance of a CSMA network.

A. Motivation for IE

The interactions between links, described in the previous section, arise as a combination of the location of the nodes, the RF characteristics of the environment as well as the radio parameters such as transmission power and Carrier Sense (CS) threshold. Typically, radio parameters are fixed by the manufacturer, giving rise to situations where destructive interactions frequently occur. Interaction Engineering controls one or more of the transceiver parameters to control the interaction modes between the competing links.

Several parameters including transmission power, CS threshold, Receiver Sensitivity (RS) threshold, transmission rate and channel selection, affect the interactions, and hence the performance of the network. Among these, we focus on the three primary parameters: *transmission power, CS and RS thresholds*. While other parameters can also be controlled to further improve the performance, the primary parameters dictate the interactions that occur between nodes for a given channel. For example, even in the presence of multiple channels, it is vital to configure the primary parameters on links that operate on the same channel; we demonstrate the need in Section VI-B by showing the performance improvement gained by tuning the parameters. Hence, configuring the primary parameters is a critical piece of the whole problem; we call the three primary parameters as *interaction parameters*.

An ideal solution is to adjust the interaction parameters of the nodes such that each link has NI interaction with other links, and all links concurrently transmit without any hidden terminals. For example, consider the simple scenario as shown in Figure 2 (a). It is easy to tune the variables such that links *A-B* and *C-D* have NI interaction. In WLAN deployments, some studies suggest that if an AP uses high transmission power it should set low carrier-sensing thresholds to maintain symmetry of the network [15], [16]. We show a contradictory observation where the symmetry policy does not necessarily lead to optimal results. In Figure 2 (a), both nodes *A* and *C*

can have a high carrier sensing threshold and relatively high transmit power as long as they do not interfere with the other destination. This results in optimal NI interaction.

However, collectively optimizing the parameters of all nodes to reach this best-case scenario is in general impossible. For example, the topology in Figure 2 (b) cannot achieve NI interaction. The receivers are close to the sender of the other link, and hence experience high interference making it impossible to decouple the links. The best interaction in such a topology is to tune the parameters such that there is an SC interaction to help the transmitters avoid hidden terminals. Hence, the topology and the parameters of all nodes collectively impact the performance, and should be carefully tuned to avoid detrimental interactions.

B. Related Work

An earlier version of this paper [11] presented the idea of interaction engineering. The current paper extends and generalizes the analytical model, evaluates and specializes the model for unmanaged and managed WLANs, and for different classes of traffic.

Several related studies have proposed optimization of CSMA network by tuning power, carrier sense and receiver sensitivity. CSMA protocols, such as IEEE 802.11, has been optimized to address various problems such as transmission power control [10], [12], [17], effective carrier sensing [7], [9], [18], [6] and avoiding the capture effect [14]. While each study solves a particular piece of the puzzle, none consider all the parameters that influence the MAC interaction. We organize existing research under four areas based on their limitations, and compare each with the IE methodology.

1) *Studies that optimize a subset of transceiver parameters:* Transmission power, carrier sensing threshold and receiver sensitivity of different nodes that govern the network behavior are inter-dependent. Some related studies consider only a subset of parameter space. For example, existing MAC-level topology control algorithms alter transmission power assuming constant carrier sensing threshold for all nodes [10], [12], [17]. Other studies alter carrier sensing assuming that transmission power is constant [19], [8], [9], [6], [18]. In addition, both these categories do not consider the effect of capture [14] due to improper receiver sensitivity.

Other models and protocols jointly tune carrier sensing and transmission power; this work is most relevant to our study [15], [16], [20]. Some approaches discuss heuristics that possibly reduce hidden and exposed terminals [20]; we provide a model and analysis of tuning CSMA protocol to eliminate all hidden terminals and minimize exposed terminals. Mhatre *et al.* adjust the transmission power and carrier sensing thresholds of the access points in an AP [15], [16]. However, the objective of the study is to reduce asymmetry. However, as we show in the examples in Section III-A, reducing asymmetry does not necessarily optimize interactions in CSMA. Moreover, while the focus of the above studies is only optimizing AP-AP interaction. We optimize any two links – both in single-hop WLAN networks and multi-hop networks.

In summary, models that capture only a part of the parameter space do not efficiently optimize the network; effective interaction engineering is achieved only when transceiver parameters of *all* nodes are jointly optimized. The paper proposes model, protocol and evaluation for effective interaction tuning.

2) *Studies focusing on random distribution of nodes:* Models in this category alter carrier sensing under specific network structure and density [6], [7]. For example, the carrier sensing models in the above studies assume a dense random network and a honey-comb structure, where transmitters are at the centers of a hexagon and interfering transmissions happen only from neighboring cells. While optimizing in such a dense network provides asymptotic bounds, it does not faithfully optimize a given arbitrary network. Such an optimization may still have hidden and exposed terminals. For example, we saw that scenario 1 and 2 in Figure 2 need specific tuning of carrier sensing to have good interactions – even if they had belonged within the same cell. In contrast, we optimize CSMA for a given arbitrary topology.

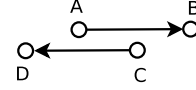
3) *Studies accounting for limited CSMA rules:* Studies that characterize the possible CSMA effects conclude that a notable fraction of the detrimental interactions are caused not only due to classical hidden terminal scenario, where a DATA packet by another DATA packet, but also due to detailed CSMA handshaking rules [4], [5].

For example, two-way handshaking requires proper parameter tuning not only at the sender but also at the receiver. Inappropriate or default values often introduces the possibility of DATA collision due to ACK from a neighboring link [5]. Moreover, the receiver does not carrier sense before transmitting ACK, and hence the same rules as DATA transmission cannot be applied at receivers. Existing models do not consider the common CSMA protocol rules such as two-way handshaking [20], [21]. Hence, the solutions are incomplete and may cause undesirable interactions. Our work proposes a framework that accounts for interaction parameters based on prominent CSMA rules.

4) *Studies optimizing contention regulation:* Some studies control the contention observed in the network by regulating the IEEE 802.11 backoff window [22], [23], [24], [25], [26]. However, they do not regulate the MAC level interactions. In contrast, IE focuses on optimizing the CSMA interactions in the network, thus leading to more effective handshakes. In future, we plan to pursue model and protocol for a joint



(a) Scenario 1: Collision to NI interaction



(b) Scenario 2: Collision to SC interaction

Fig. 2. Interaction Engineering to prevent collisions

contention regulation and interaction optimization.

In summary, CSMA optimization is a joint optimization of several inter-dependent parameters belonging to several nodes. CSMA interactions between the links have to be optimized to maximize spatial reuse and avoid detrimental effects. However, existing studies account for only a subset of the effects and parameters, and do not explicitly consider all CSMA interactions. In this paper, we propose a generic methodology for optimizing the interactions in CSMA by tuning parameters such that the links have most favorable interactions between them.

IV. LINK-PAIR ENGINEERING

In this section, we formulate an optimal IE model for a pair of links. Then, we propose a centralized algorithm that extends the model for a general topology with n links.

A. Strategies to optimize CSMA

We start by providing the main intuition for IE using simple scenarios. We first discuss engineering detrimental interactions that suffer collisions (AIS, SIS and IDIS interactions). We then discuss avoiding the capture effect (HTC interactions).

Consider the scenario in Figure 2 (a), where link $A-B$ can have the several interactions with $C-D$ based on the the interaction parameters. Packet collisions at link $C-D$ can be avoided either by: (i) raising the transmission power at $C-D$ or lowering power at $A-B$ such that two links transmit concurrently, but do not experience collision (having NI interaction); or (ii) Creating an SC interaction between the links by altering CS threshold at A and C . Engineering the topology for NI interactions (option i), is better since it achieves higher spatial reuse than SC interaction.

However, in certain scenarios such as Figure 2 (b), tuning only the transmission power does not eliminate collisions. Increasing transmission power of $A-B$ increases the interference at link $C-D$, causing collisions at $C-D$. In response, $C-D$ increases its transmission power, causing greater interference at $A-B$, and the cycle continues without eliminating collisions.

CSMA also suffers from HTC interactions (those with capture). Recall that capture occurs when the destination first “locks on” to the packet from the interferer, and thus fails to lock to the stronger signal from source. This effect can be eliminated by setting the RS threshold high enough to prevent locking to the weaker interfering signal.

IE requires the given scenario to be analyzed to identify harmful interactions. We can then attempt to convert these interactions to NI or, failing that, to SC interaction by altering transmission power or CS threshold. HTC interactions are then prevented by setting RS threshold appropriately.

B. Problem statement and network representation

We define optimal IE as a problem that eliminates all destructive interactions (AIS, SIS, IDIS) by converting them to either NI or SC. We define *favorable interactions* as NI and SC (NI being more favorable than SC), since (i) NI and SC are the only two interactions with *successful handshakes* in the CSMA protocol; and (ii) NI achieves greater capacity than SC (Figure 2).

Network topology is represented with a set of nodes \mathcal{N} . One-hop traffic between the source nodes S_i and respective destination nodes D_i is represented by link set $\mathcal{L} = (S_i, D_i)$. The number of nodes and active links are denoted by N and L , respectively. We assume the Two-ray Ground propagation model, where the signal received at a node b when a is transmitting at power p_a is given by:

$$S_{ab} = \frac{G_t G_r h_t^2 h_r^2}{d_{ab}^\theta L} p_a = C_{ab} p_a, \quad (1)$$

where d_{ab} is the distance between a and b , G_t and G_r are antenna gains of transmitter and receiver, h_t and h_r are antenna heights, L is the system loss, and θ is the propagation constant. We simplify the equation by denoting the constant terms by C_{ab} . The *Signal to Noise Interference Ratio* (SINR) interference model is assumed, where the packet is received without errors if the ratio of signal to the noise and interference is greater than a threshold β .

IE can be modified to work with alternative propagation models. Even though the two-ray ground model is assumed in analytical calculations, our testbed experiments conclude that it is reasonable to approximate the received power to be proportional to the transmission power.

C. Optimal model for two links

We now formulate an *Optimal Link Pair* (OLP) model for link-pair (s_1, d_1) and (s_2, d_2) . We define this model as $OLP((s_1, d_1), (s_2, d_2))$. The model first checks if the topology can be configured for an NI interaction. If it is infeasible, we set the CS threshold such that both links do not concurrently transmit.

Step 1: Check for NI feasibility: If two links can be engineered to have NI interaction, then the SINR for each link should be greater than SINR threshold β , i.e.

$$\frac{S_{s_1 d_1}}{W + S_{s_2 d_1}} \geq \beta,$$

where W is the background white noise. Similarly, NI also dictates that (1) ACK from (s_2, d_2) should not corrupt DATA of (s_1, d_1) (to avoid IDIS interactions); and (2) DATA from a link (s_2, d_2) should not corrupt reception of ACK at (s_1, d_1) ¹.

¹Another possibility of collision is ACK of (s_2, d_2) corrupting ACK of (s_1, d_1) . While this constraint can be easily added to the model, we ignore it since we have observed that the chances of two ACKs being sent at the same time is very low (around 3% in our simulations).

Hence, using Equation 1, constraints for NI can be expressed as three linear constraints for each link (s_1, d_1) :

$$\begin{aligned} p_{s_1} C_{s_1 d_1} - \beta(p_{s_2} C_{s_2 d_1}) &\geq \beta W, \\ p_{s_1} C_{s_1 d_1} - \beta(p_{d_2} C_{d_2 d_1}) &\geq \beta W, \\ p_{d_1} C_{d_1 s_1} - \beta(p_{s_2} C_{s_2 s_1}) &\geq \beta W. \end{aligned} \quad (2)$$

Similarly, three constraints are expressed for link (s_2, d_2) . Further, the variables transmission power p_i , CS threshold c_i and RS threshold r_i for node i are bounded by

$$p_i \geq P_{\min i}, \quad c_i > W, \quad \text{and} \quad r_i > W, \quad (3)$$

where W is the white noise and $P_{\min i}$ is the minimum transmission power for node i .

Equation 2 represents the constraints for the feasible region of transmission powers of the nodes. We now present the feasible regions for RS threshold and CS threshold of the nodes. If NI is feasible, the CS threshold should be set such that s_1 and s_2 do not sense each other. The RS threshold at the nodes should be set such that it receives the packet from its link, but does not lock to the DATA packet from other link (to avoid HTC interaction). These constraints are represented by:

$$\begin{aligned} c_{s_1} &> p_{s_2} C_{s_2 s_1} + W, \\ p_{s_2} C_{s_2 d_1} + W &< r_{d_1} \leq p_{s_1} C_{s_1 d_1} + W, \\ r_{s_1} &\leq p_{d_1} C_{d_1 s_1} + W. \end{aligned} \quad (4)$$

We formulate the parameter constraints for (s_2, d_2) in a similar way. The feasible region of the interaction parameters is defined by the constraints in Equations 2, 3, 4 and similar equations for (s_2, d_2) . If the above constraints are feasible, the NI interaction is possible, and we choose the interaction parameters from the feasible space. Otherwise, it is impossible to create an NI, and hence we prevent concurrent transmission of the links by creating SC interaction.

Step 2: SC formulation: The constraints for SC are similar to the ones in NI, except for: (i) CS threshold should be set such that s_1 and s_2 sense each other, and hence do not transmit concurrently (Equations 7), and (ii) it is sufficient if the SNR (instead of SINR) is greater than β , since the links are separated in time and there is no interference from the other link (Equations 5, 6). They are given by constraints:

$$p_{s_1} C_{s_1 d_1} \geq \beta W, \quad (5)$$

$$p_{d_1} C_{d_1 s_1} \geq \beta W, \quad (6)$$

$$c_{s_1} \leq p_{s_2} C_{s_2 s_1} + W, \quad (7)$$

$$r_{d_1} \leq p_{s_1} C_{s_1 d_1} + W, \quad (8)$$

$$r_{s_1} \leq p_{d_1} C_{d_1 s_1} + W. \quad (9)$$

Similar constraints are set for link (s_2, d_2) .

In summary, OLP first checks feasibility of NI interactions and assigns interaction parameters if it is feasible (step 1). Otherwise, we create SC interaction by following step 2. Thus, we optimally engineer any two-link topology by maximizing favorable interactions while avoiding detrimental interactions.

Theorem IV.1. (Feasibility of OLP) $OLP((s_1, d_1), (s_2, d_2))$ is always feasible

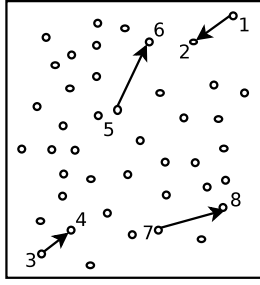


Fig. 3. A 4-link scenario

Proof: Consider a pair of links (s_1, d_1) and (s_2, d_2) . Step 1 of the OLP (Equations 2 to 4) attempts to find feasible transceiver parameters such that the links have NI interaction. There are two possibilities:

- (1) Step 1 is feasible in which case respective transceiver parameters are assigned.
- (2) Step 1 is infeasible, in which case we solve for Step 2 to check if the links have SC interaction.

We now show that Step 2 is always feasible. If we set the sensing threshold (c) of sources above noise (W), and set transmission power p of the nodes such that the received signal strength at the other receiver is above the c , then the two links will have an SC interaction. There are no constraints in Step 2 that restricts the above feasibility. Hence, SC interactions is always feasible between a pair of links. ■

Corollary IV.2. (Lower Bound on Max Power) *In OLP($(s_1, d_1), (s_2, d_2)$), if the transmission powers at the sources s_1 and s_2 are greater than some maximum transmission power, then the links can always be configured into an SC interaction, i.e., SC interaction is always possible for the below transmission power regions of the sources:*

$$\begin{aligned} p_{s_1} &\geq P_{\max_{s_1}}((s_1, d_1), (s_2, d_2)) & \text{and} \\ p_{s_2} &\geq P_{\max_{s_2}}((s_1, d_1), (s_2, d_2)), \end{aligned} \quad (10)$$

where $P_{\max_s}(l_1, l_2)$ is the lower bound on maximum transmission power for SC interaction for source s while optimizing link pair l_1 and l_2 . $P_{\max_s}(l_1, l_2)$ can be calculated from constraints of Step 2. For example, applying Equations 3, 5 and 7, after manipulation yields $P_{\max_{s_1}}((s_1, d_1), (s_2, d_2)) = \max(\frac{\beta W}{C_{s_1 d_1}}, \frac{1}{C_{s_1 s_2}})$.

The above corollary can be directly derived from Theorem IV.1

D. Centralized Link Pair Engineering (CLP): An Approximate model for n -link topology

We now extend the OLP model which works with a pair of links to a general heuristic algorithm for n links; we call this Centralized Link Pair (CLP) algorithm. CLP iteratively optimizes all link pairs. Each link calculates the optimal interaction parameters with other links, and updates its bounds on the interaction parameters. Our algorithm eliminates all hidden terminals, while attempting to maximize NI interactions.

We illustrate the algorithm through an example topology in Figure 3. In the optimal solution, all links can be configured

Algorithm 1 Centralized Link-Pair (CLP) Algorithm

```

1: //lb[n] = Lower bound on tx power for node n
2: //S[l] = Set of links that have SC interaction with link l
3: Initialize lb[n] = 0 and S[l] = ∅
4: repeat
5:   converged = false;
6:   for all link-pairs  $(s_1, d_1), (s_2, d_2)$  do
7:     Set  $P_{\min_i} = \text{lb}[i]$  for  $i \in s_1, d_1, s_2, d_2$ 
8:     Run OLP  $((s_1, d_1), (s_2, d_2))$ 
9:      $p_{s_1}, p_{d_1}, p_{s_2}, p_{d_2} = \text{Min tx power from OLP.}$ 
10:    Update  $\mathcal{S}[(s_1, d_1)]$  and  $\mathcal{S}[(s_2, d_2)]$ 
11:     $\text{lb}[s_1] = \max(\text{lb}[s_1], p_{s_1})$ 
12:    Similarly update  $\text{lb}[d_1], \text{lb}[s_2], \text{lb}[d_2]$ .
13:   end for
14:   if lb and S has not changed then
15:     converged = true;
16:   end if
17: until converged
18: for all link  $(i, j)$  do
19:    $p_i = \text{lb}[i]; p_j = \text{lb}[j]$ 
20:    $c_i = c_j = \text{Maximum CS threshold}$ 
21:   for all link  $(x, y) \in \mathcal{S}[(i, j)]$  do
22:      $\text{newCs} = p_x C_{xi} + W$ 
23:     if  $\text{newCs} < c_i$  then
24:        $c_i = \text{newCs};$ 
25:     end if
26:   end for
27:    $r_i = p_j C_{ji} + W; r_j = p_i C_{ij} + W$ 
28: end for
29: return  $p, c, r$ 

```

to have NI interactions with each other except the link-pair 1-2 and 5-6, which have to be configured for SC.

Our approach is described in Algorithm 1. We run the OLP on each link pair. Each solution to the OLP computes a feasible range for interaction parameters, thus providing $n - 1$ feasible regions. For example, the interaction parameters for 1-2 has three feasible regions (one with each other link-pair).

The solution converges to the optimal if the feasible regions intersect. Interaction parameters for a node chosen from the intersecting region satisfies all constraints. However, if the feasible regions do not intersect, then we use a heuristic.

The heuristic uses a method of optimizing transmission power first, and then assigning the other two interaction parameters, instead of jointly optimizing all the three interaction parameters. The intuition is to select the minimum feasible transmission power for a node, since such a choice leads to greater spatial re-use (higher possibility of NI). The minimum feasible power for a node i is recorded in $\text{lb}[i]$. Similarly, we also record the links with which a given link l has SC interaction in the set $\mathcal{S}[l]$.

The recorded lb becomes the new lower bounds in the next iteration of all link pair optimization. The iterations continue until there is no change in lb and \mathcal{S} . Then we calculate the CS and RS threshold.

Before we prove CLP convergence, we define some prop-

erties necessary for the proof. We extend the lower bound on transmission power while optimizing link-pair $(P_{\max_i}(L_1, L_2)$ in Equation 10) to a lower bound for a particular node (P_{\max_i}) .

Definition (*Per-Node Lower Bound of Tx Power for SC interaction – P_{\max_i}*): Let L_i be set of all links where i is either a source or destination. We define the lower bound of transmission power for a node i such that the links that have node i as source is in SC interaction with all other links as

$$P_{\max_i} = \max(\{P_{\max_i}(L_i, L'), \forall L' \in \{L - L_i\}\}), \quad (11)$$

where L is the set of all links.

Lemma IV.3. (Non-decreasing lb) *The minimum transmission power for a node i provided to the OLP algorithm $(\text{lb}[i])$ is a non-decreasing function across successive iterations in Algorithm 1.*

Proof: The OLP(l_1, l_2) algorithm is evaluated at each loop iteration between each link-pair l_1 and l_2 (line 8) with minimum transmission powers set as per lb computed in the previous loop (line 7). At each loop, $\text{lb}[i]$ is updated as the maximum value of previous $\text{lb}[i]$ and the new transmission power as per OLP result (lines 11 and 12). Hence, $\text{lb}[i]$ is non-decreasing function across multiple loop iterations. ■

Theorem IV.4. (CLP convergence) *CLP Engineering always converges.*

Proof:

In order for the CLP algorithm to converge, the repeat loop (from lines 4 to 17) should terminate. The rest of the algorithm computes other transceiver values, and is guaranteed to return. Lemma IV.3 shows that lb is non-decreasing. Hence, there are two cases to evaluate:

Case 1: If all nodes have the same $\text{lb}(t+1)[i]$, then the algorithm converges (line 15).

Case 2: If any of the node has a higher transmission power, i.e. if $\text{lb}(t+1)[i] > \text{lb}(t)[i]$ for some i , then the loop is repeated. Based on the value of $\text{lb}(t+1)[i]$, there are two subcases:

Subcase 2.1: $\text{lb}(t+1)[i] \leq P_{\max_i}$: In this case, the OLP algorithm in $(t+1)$ iteration may choose an equal or greater value of $\text{lb}(t+2)$. This leads to repeated evaluation of Case 1 and 2.

Subcase 2.2: $\text{lb}(t+1)[i] > P_{\max_i}$: Here we are assured that the OLP algorithm in the $(t+2)$ iteration will not increase the value of $\text{lb}[i]$ in any successive iterations since SC interaction has already been achieved with all other links (Equation 11). Hence, $\text{lb}[i]$ will be constant from iteration $(t+1)$.

By symmetry, the argument applies to all nodes i : it can be shown that the iterations may loop only until $\text{lb}[i]$ reaches a transmission power greater than P_{\max_i} . After this stage, there will be no increase in lb, and hence the algorithm converges (line 15) ■

CLP does not consider cumulative interference from multiple links. Considering cumulative interference substantially increases the complexity of the problem, preventing practical solutions. We protect against cumulative interference by using conservative estimates of the SINR threshold β (in our experiments, we use a safety margin of 1.2). Note that cumulative

interference occurs from NI links, and that the algorithm protects against the NI link that causes the most interference; by having a safety margin, we consider concurrent interference from links other than the bounding one as part of noise.

V. INTERACTION AWARE MAC

In this section, we develop the distributed *Interaction Aware MAC* (I-MAC) protocol which is based on the developed centralized model (CLP). Starting from CLP where the link-pair problem is solved iteratively (Section IV), I-MAC exchanges information about neighborhood links, and locally optimizes only with those links. We show that such a protocol performs close to the centralized solution and can adapt to dynamism such as connection initiation or termination.

Each node i maintains its lower bound on transmission power lb_i , in addition to the already existing variables for transmission power (p_i), CS threshold (c_i) and RS threshold (r_i). The information about the links is maintained in two tables: self-links are maintained in Link Information Table (LIT) and neighboring link information is maintained in Neighboring Link Table (NLT).

LIT is maintained at the source of a link. Each row consists of the tuple:

$$\langle d, \text{lb}_d, C_{sd}, C_{ds}, \text{numPktsSent}, \text{lastUpdateTime} \rangle,$$

where d is the destination id.

NLT is maintained by all nodes. Each row contains

$$\langle s_2, d_2, \text{lb}_{s_2}, \text{lb}_{d_2}, \text{scNeighbor}, \text{ttl}, \text{lastUpdateTime} \rangle,$$

where s_2 and d_2 are the source and destination of the neighboring link. The boolean variable scNeighbor indicates if the neighboring link should be configured as SC (e.g. Figure 2 (b)). The NLT is propagated to the neighbors if the time-to-live $\text{ttl} > 0$, and the time when entry was last updated is stored in by lastUpdateTime .

Nodes initiate *Link Measurement Protocol* to measure the propagation constant (C_{ab} in Equation 1). In our testbed, we perform link measurements [27] to calculate this variable. Such algorithms have $O(N)$ complexity to measure all propagation factors in a network [28], and can be invoked on demand.

A. Protocol

The source node updates the numPktsSent in LIT dynamically as it transmits each packet. We consider a source as *active source* if numPktsSent is greater than a threshold for some link in LIT.

Each node maintains an *epoch timer* that is triggered at constant time intervals. At the end an epoch, each active source triggers two procedures:

- (i) It requests the lower bound of all its destinations (lb_d), and updates the LIT. Hence, all the fields in LIT are updated.
- (ii) It broadcasts *Link Broadcast Packet*, which consists of the following information:

$$[\text{numLinks}, \text{LinkInfo}_1, \text{LinkInfo}_2, \dots],$$

where link information LinkInfo_n contains the source, destination and their lower bounds on transmission power ($[s, d, \text{lb}_s, \text{lb}_d]$) of active links.

Upon reception of the Link Broadcast packet, all active sources update their NLT. If there is a change in the NLT, then the active source initiates the calculation of interaction parameters. It runs the Algorithm 1 for the current links of the source with the links in NLT. The procedure provides interaction parameters (p, c, r) and new lower bounds on transmission power (lb) for all links. The source updates its lb and interaction parameters, and requests the destination to set the new interaction parameters. This procedure repeats until the lb of all links have converged.

B. Disseminating link broadcast packets

As discussed earlier, each source sends link broadcast packet to its neighbors. The broadcast is transmitted at a power such that all interferers are reached. Currently, we transmit the broadcast packet at larger power (3 dBm more than the default power) since it has been observed that most of the interferers are reachable with such an improvement [29]. We also set the ttl of the link broadcast packet to 3 such that most interferers are reachable through multiple hops.

Another design aspect of the current protocol is the reliable dissemination of broadcast. Once the interaction parameters of the nodes have been altered, they are configured to receive the unicast packets only from their links. Hence, the nodes might fail to intercept broadcast packet if their receiver sensitivity is set high. We implement a scheme of sending the link broadcast packets through a control channel. In our evaluation, we show the effectiveness of I-MAC in both single-channel and two channel scenarios.

VI. PERFORMANCE EVALUATION

In this section, we evaluate the performance of the proposed model, centralized algorithm and I-MAC protocol. First, we demonstrate the effectiveness of IE in two link scenarios using simulation. We then analyze the results in general n link topologies – both in Wireless LANs (WLANs) and multi-hop networks. Finally, we evaluate the performance of OLP in a testbed. We use QualNet [30] for simulation studies, where operate on IEEE 802.11b with 2Mbps channel capacity. In scenarios which require high rate (such as video traffic), we use 11 Mbps channel capacity.

A. Two-link Topologies

We evaluate the effectiveness of the optimal two-link model (OLP and CLP) as well as I-MAC through simulations. We simulate 400 scenarios of two links which are randomly chosen in a 1.5 sq km area. Figure 4(a) shows the interactions in IEEE 802.11b (standard case) that have been converted to more favorable interactions in I-MAC. The SC-NI conversion represent the exposed terminals, where links can be concurrently scheduled without any bad interactions. Around 34% of the scenarios suffer from exposed terminals, and all have been successfully converted to NI interaction in I-MAC. Similarly,

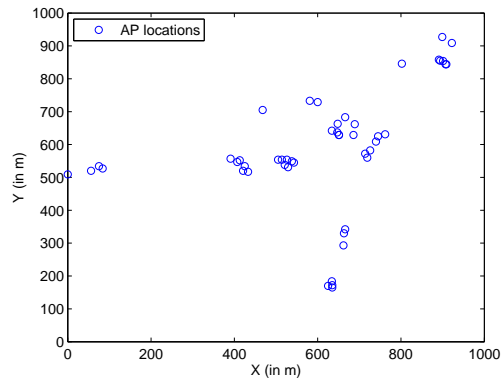


Fig. 5. WLAN topology: 49 APs spread in an area of 1 sq km at Dartmouth University [32]

nearly 20% of the scenarios suffer from packet timeouts (e.g., AIS, IDIS, HTC), which have all been transformed to NI or SC interactions.

The gain in throughput for each conversion is shown in Figure 4(b). It can be seen that destructive interactions (those that result in ACK timeouts) drastically affect the throughput, sometimes even reducing the link throughput to zero. I-MAC converts all the interactions to high throughput interactions. Throughput is doubled in exposed terminal scenarios as the unnecessarily blocked link is allowed to transmit.

Figure 4(c) shows the cumulative network throughput improvement in different scenarios. Cumulative network throughput improves in 60% of the scenarios, with maximum throughput improvement of 350%. The CLP and I-MAC protocol converge to the optimal solution in two-link scenarios. System aspects such as unreliable broadcast account for around 1% of the scenarios whose performance slightly degraded.

B. Interaction Engineering in WLANs

We show the effectiveness of IE in WLANs by controlling the transceiver parameters of base-stations and clients. In the WLAN scenario, we modified the CLP algorithm such that clients and APs collect the interference parameters from other nodes. All the clients relay the information to the APs. The APs coordinate over the back-end wired network to a centralized server that executes the CLP algorithm and announces the new transceiver parameters to the nodes. Currently, the above algorithm requires modification on APs and clients; the nodes have to be equipped with measurement protocols to estimate interference parameters. In future, we plan to evaluate intelligent and distributed measurement techniques where the APs sniff client packets, and infer the interference parameters (similar to micro-probing [31] or LQ-measure [27]).

We use the WLAN topology at Dartmouth University [32]. The WLAN consists of 49 APs located on the first floor of different university buildings in a 1 sq km area (Figure 5). We demonstrate the network performance with CLP in a WLAN where all APs operate on the same channel.

We first demonstrate the performance of CLP in unmanaged WLANs, such as home-networks, where APs may reside on

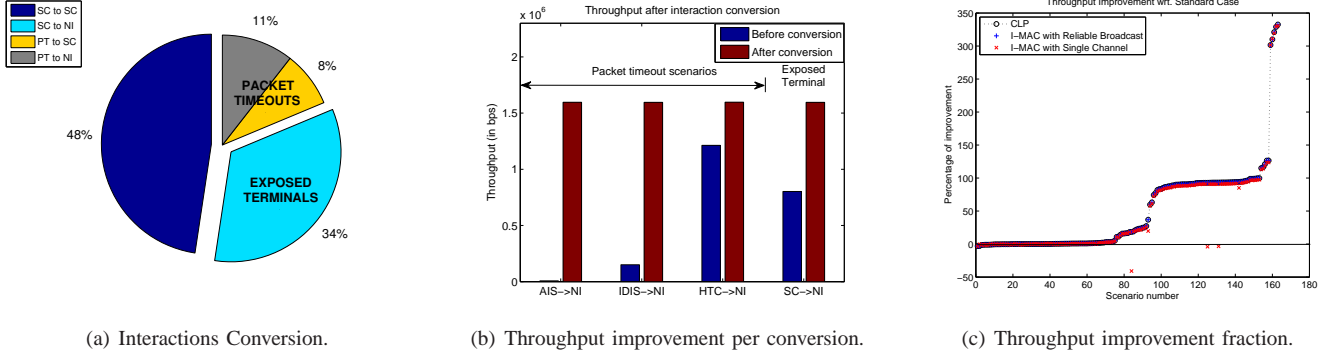


Fig. 4. 2-Link topologies: (a) All exposed terminals (34% scenarios) are converted to NI; hidden terminals (denoted by PT) are converted to either NI or SC. (b) Conversion into NI interactions lead to large throughput gains. (c) Overall network throughput improvement as much as $3.5\times$ is observed by I-MAC protocol.

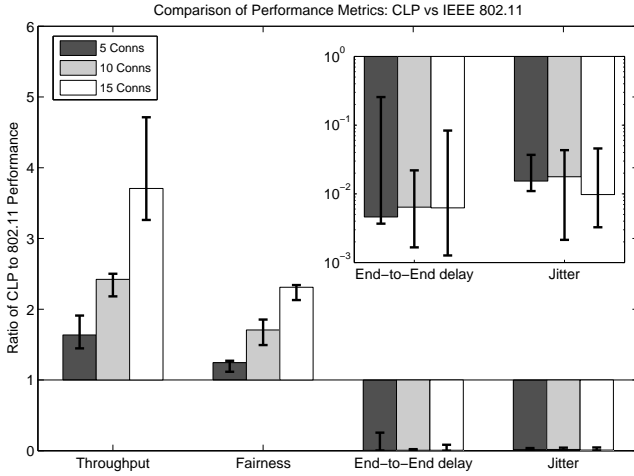


Fig. 6. WLAN under CBR traffic: CLP achieves $4\times$ more throughput, $2.5\times$ more fairness, and more than two orders of magnitude less end-to-end delay and jitter

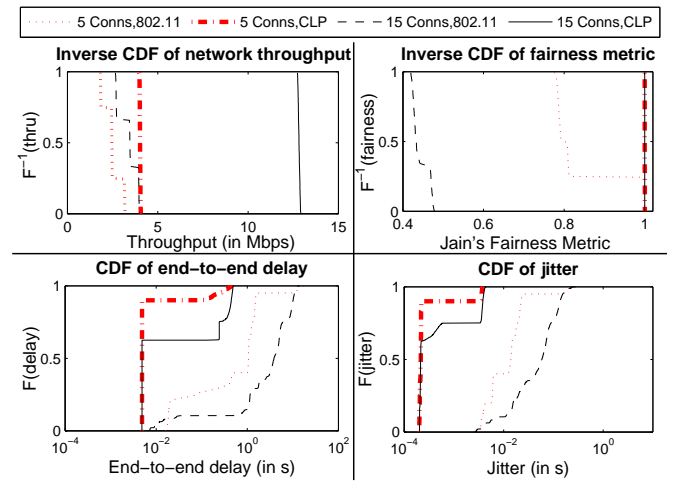


Fig. 7. WLAN under CBR Traffic: Cumulative distribution functions of network metrics in a WLAN

the same channel. We study the performance of CLP under different types of traffic such as CBR, TCP and video traffic.

Performance under saturated CBR traffic:

We vary the load on the network by altering the number of connections, and under each load we simulate 160 scenarios. Figure 6 shows the improvement in the network throughput, fairness, end-to-end delay and jitter. X-axis plots different performance parameters under different loads. Y-axis represents the median values of the performance with error-bars representing the 25% and 75% quartiles. Fairness is measured by Jain's fairness metric [33]. Figure 6 compares performance metrics (throughput, fairness, delay and jitter) in CLP and 802.11. Y-axis indicates the ratio of the performance metric observed in CLP to that of 802.11; a value of n on the Y-axis implies that the performance metric of CLP is n times than the metric observed in 802.11. We observe an overall network throughput improvement of $4\times$ under high loads. A fairness improvement of $2.5\times$: eliminating destructive interactions substantially improves fairness, although other sources of unfairness, such as contention unfairness [24], remain. As highlighted in the inset figure, the end-to-end

delay and jitter shows more than two orders of magnitude improvement.

Figure 7 plots the inverse Cumulative Distribution Function (Inverse CDF) of the observed throughput and fairness as well as the CDF of the delay and jitter. CLP provides a median throughput (throughput at $F^{-1}(thru) = 0.5$) that is $3.7\times$ greater than IEEE 802.11 with a near-perfect fairness for a scenario with 15 connections. The CDFs of end-to-end delay and jitter show an order of magnitude improvement (X-axis is plotted in log scale).

Performance under TCP: Figure 8(a) shows the superior TCP performance of CLP over IEEE 802.11. For example, in the 20 connection scenario, CLP provides approximately $6\times$ times throughput improvement with almost perfect fairness when compared to 802.11.

We observed that most of the TCP connections under 802.11 have large connection initiation delays. For example, the median connection initiation delay was 0.8 seconds under 802.11 (with a maximum delay of around 3 minutes). The large initiation delay for some connections under the 802.11 protocol is because of the inability of the connections with

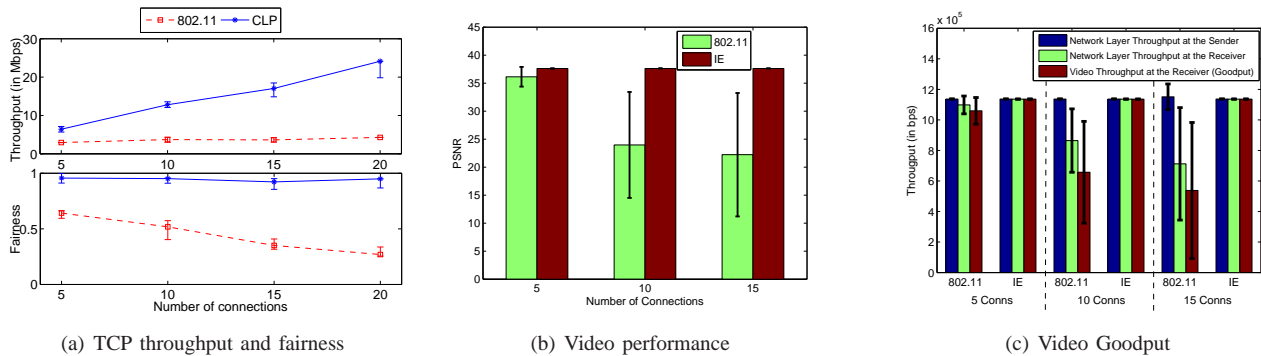


Fig. 8. TCP and video traffic over WLAN: CLP boosts performance of TCP throughput and video PSNR by reducing the packet collisions and increasing concurrent transmissions

packet collisions to contend with other connections that have effective interactions. CLP eliminates these detrimental interactions, and reduces the connection initiation delay. We observed a median delay of 0.18 seconds, and a maximum delay of 0.44 seconds under CLP.

Performance under video traffic: We added a real-time video encoding and decoding application to the QualNet simulator using *ffmpeg* and *libavcodec* library [34]. We simulated down-link video connections transmitting an 1 Mbps MPEG video of standard video benchmark (Foreman sequence) [35] with 25 frames per second. We fix the channel capacity to 11 Mbps, and vary the number of clients in the WLAN.

Figure 8(b) shows the performance of CLP under video traffic. We measure the video quality by the video Peak Signal-to-Noise Ratio (PSNR) metric. Similar to Signal-to-Noise Ratio, higher PSNR indicates better video quality. Usually, a PSNR of above 30 is considered good for compressed videos such as MPEG. Figure 8(b) shows that the PSNR of the received video with 802.11 protocol decreases below acceptable limits (around 20 dB) when the number of connections increase. CLP maintains a good PSNR with low standard deviation even under heavy traffic.

The reduction in the PSNR in 802.11 occurs because of packet losses due to collision or congestion. Video traffic is made of several frames. Some frames are large and are transmitted as a sequence of packets (as high as 18 packets per frame in our experiments). The frame cannot be rendered correctly even if a single packet in the frame is lost. Hence, there is a difference in the network throughput and perceived video goodput. Network throughput at the sender and receiver is the rate at which the data is sent or received at the network layer. Video goodput is the rate at which the successful frame data is being received. Figure 8(c) shows the significant degradation of video goodput in standard 802.11 and the effectiveness of preventing collisions by the CLP algorithm. In 802.11, only a fraction of the sent packets are received at the network layer at the receiver, out of which only a fraction is translated into successful frames. We observed that standard 802.11 realizes only 40% (median value) of the sender throughput as video goodput under high video traffic loads; CLP realizes almost 100% of the throughput as video goodput.

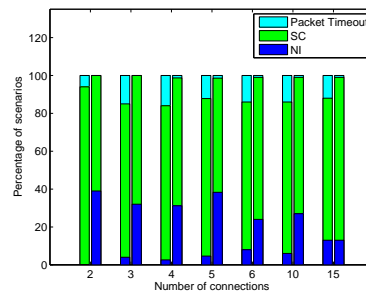


Fig. 9. Interaction frequency: The two grouped bars indicate the interactions under standard case and I-MAC, respectively. Under default configuration, packet timeouts increase as the number of connection increases. I-MAC eliminates packet timeouts and exposed terminals.

C. Interaction Engineering in Multi-hop Networks

We evaluate I-MAC protocol in both single-link topologies and multiple hop chains in multi-hop networks.

1) *Single-hop links:* We study the effectiveness of I-MAC protocol in n -link scenarios. We first study its ability to convert interactions and then evaluate performance metrics. We simulate IEEE 802.11 (standard case) and I-MAC while varying the packet sending rates and number of connections in a 1.5 sq km network.

Figure 9 shows the number of interactions in the standard case and I-MAC as the number of connections are varied. In the standard case, the number of packet timeouts stabilize at around 14% as the number of connections increase. I-MAC eliminates almost all packet timeouts. The exposed terminals that are converted are indicated by the difference between the SC scenarios in standard case and the ones in I-MAC.

As expected, the number of connections are increased, the number of exposed terminals are reduced as there is higher contention for the medium. If the link converts SC to NI with some nearby link, the CS threshold may induce packet timeout with another farther link. I-MAC prevents such conversions.

Figure 10 shows the effect of I-MAC on overall network throughput in different scenarios for varying numbers of connections. As the number of connections increases, detrimental interactions occur more frequently. Hence, I-MAC achieves improvement in larger number of scenarios as we increase the

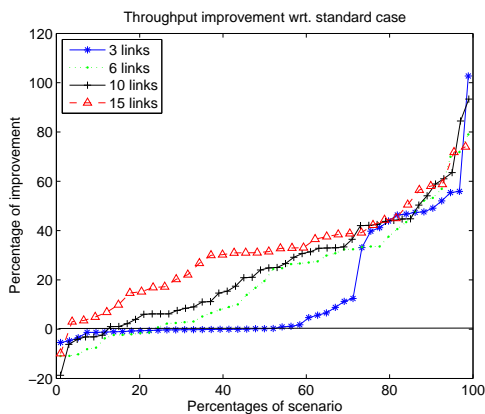


Fig. 10. Throughput improvement in n -link topologies: As the number of connections increase, larger number of scenarios have improvement with I-MAC.

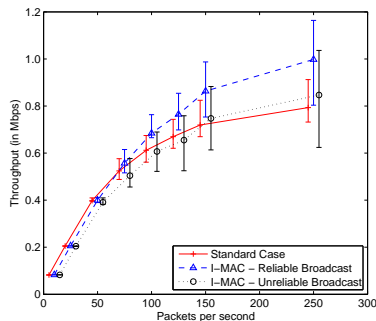


Fig. 11. Throughput as a function of traffic intensity: I-MAC with control channel to disseminate broadcast packets is significantly better than standard case and I-MAC without reliable broadcast.

number of connections. Maximum improvements of up to $2\times$ of overall network throughput is observed.

We now evaluate the throughput and delay metrics as the packet sending rate increases. We altered packet sending rates of connections in a 6-link topology. Figure 11 shows the per-connection throughput for different protocols. An average throughput improvement of around 30%. I-MAC with unreliable broadcast suffers in almost all the metrics, signifying the need for reliable dissemination of broadcast in I-MAC protocol. Improving the basic I-MAC with reliable broadcast is a part of our future work.

2) *Multi-hop Chains*: Thus far, we evaluated the effect of IE on single links. IE can be directly applied to chain topologies by considering individual links.

Figure 12 shows the performance of I-MAC protocol on multiple hop route with TCP traffic. We simulate the same two-chain topologies as above albeit with TCP traffic. Figure 12(a) show the number of scenarios with packet drops due to exceeding the retransmission limit in 802.11. The scenarios with packet timeouts drastically decreases under the I-MAC protocol.

As observed in CBR traffic, good interactions do not directly translate into better overall network throughput. Figure 12(b) show the inverse CDF of throughput under 802.11 and I-MAC

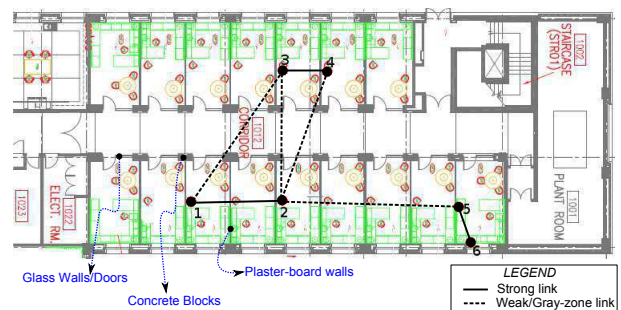


Fig. 13. Testbed

#	Links	Throughput (\pm 95%CI)		Interaction	
		Std	CLP	Std	CLP
1	1 - 2	2.912(\pm 0.028)	5.042(\pm 0.170)	SC	NI
	4 - 3	2.955(\pm 0.036)	5.296(\pm 0.009)	SC	NI
2	4 - 3	4.033(\pm 0.116)	5.299(\pm 0.009)	HTC	NI
	6 - 5	0.000(\pm 0.000)	5.292(\pm 0.012)	AIS	NI
3	2 - 1	2.852(\pm 0.098)	2.752(\pm 0.136)	SC	SC
	3 - 4	3.024(\pm 0.130)	2.804(\pm 0.092)	SC	SC
4	1 - 2	3.040(\pm 0.027)	3.263(\pm 0.070)	SC	SC
	3 - 4	2.645(\pm 0.030)	3.485(\pm 0.102)	SC	SC
5	2 - 1	2.684(\pm 0.035)	3.036(\pm 0.132)	SC	SC
	4 - 3	3.016(\pm 0.037)	3.513(\pm 0.106)	SC	SC
6	2 - 1	2.410(\pm 0.029)	4.973(\pm 0.059)	SC	NI
	6 - 5	3.678(\pm 0.027)	3.796(\pm 0.213)	HTC	HTC
7	2 - 1	2.601(\pm 0.026)	2.697(\pm 0.075)	SC	SC
	5 - 6	3.111(\pm 0.024)	3.823(\pm 0.050)	SC	HTC
8	1 - 2	1.478(\pm 0.055)	2.587(\pm 0.165)	IDIS	SC
	6 - 5	5.253(\pm 0.019)	5.287(\pm 0.014)	NI	NI
9	1 - 2	3.503(\pm 0.162)	5.251(\pm 0.010)	SC	NI
	5 - 6	5.293(\pm 0.009)	5.293(\pm 0.013)	NI	NI
10	3 - 2	0.122(\pm 0.061)	3.238(\pm 0.267)	AIS	SC
	6 - 5	5.298(\pm 0.009)	5.169(\pm 0.086)	NI	NI
11	3 - 2	0.101(\pm 0.037)	0.000(\pm 0.000)	AIS	AIS
	5 - 6	5.293(\pm 0.011)	5.230(\pm 0.003)	NI	NI
12	2 - 3	0.024(\pm 0.020)	0.000(\pm 0.000)	AIS	AIS
	5 - 6	4.910(\pm 0.208)	5.263(\pm 0.011)	NI	NI
13	2 - 3	3.010(\pm 0.269)	0.000(\pm 0.000)	SC	AIS
	6 - 5	2.997(\pm 0.041)	5.296(\pm 0.010)	SC	NI

TABLE I
CLP ON TESTBED: SCENARIOS 1-5 SHOW THE CLASSICAL CONVERSIONS. SCENARIOS 6-9 SHOW PARTIAL PACKET DROPS. CLP FAILS WHEN THE LINKS ARE TOO WEAK (# 10-13).

protocol. However, the I-MAC converts the scenarios to be more fair. The inverse CDF of Jain's fairness metric is shown in Figure 12(c). Note that the minimum value of Jain's fairness metric in a scenario with 2 connections is 0.5 (assuming at-least one connection has non-zero throughput). Figure 12(c) shows that 802.11 suffers from significant TCP unfairness; the unfairness is eliminated using I-MAC.

D. Initial Testbed Evaluation and Discussion

We now discuss the effectiveness of OLP with our initial evaluation on a testbed. Our testbed consists of six Soekris boards with Atheros chipset and modified MadWifi driver [36], operating on IEEE 802.11a with 6 Mbps. The testbed is placed in an office environment with 2 rows of rooms, as shown in Figure 13. We have observed that links that cross the rows suffer from lower received signal strength because of the concrete and glass separation. This leads to 4 reasonably

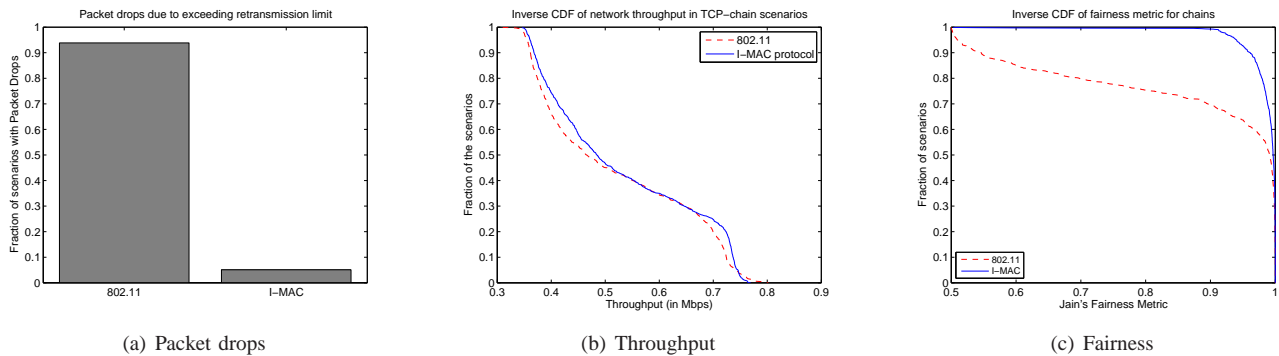


Fig. 12. TCP Chains

effective unicast links: 1-2, 3-4, 5-6, and 2-3. The CLP algorithm runs on a central coordinator, and initiates *iperf* at the nodes to measure the throughput.

All the boards are connected through an Ethernet control channel to initiate *Link Measurement Protocol* [27] and transmit link broadcast packets. The measurement protocol is initiated at the start of the experiment, where all nodes compute propagation factors C_{ij} .

The measurement protocol is limited by existing off-the-shelf cards and drivers in two ways:

1. The card does not permit altering carrier sensing and capture thresholds. Hence, we limit the CLP to tune only transmit power.
2. The protocol cannot measure fine-granularity connectivity information between all pairs of nodes. C_{ij} can be computed only at nodes which can successfully transmit some packets.

We evaluate different pairs of links and the results are shown in Table I. It can be seen that CLP optimizes certain link pairs (Scenarios 1-9), while other link pairs perform worse (Scenarios 10-13). It is not possible to identify the root-cause of the problem because of the hardware and driver limitations. However, we conjecture on why CLP optimizes only certain set of links based on some observations.

We classify the links into three categories based on the connectivity [27]:

- 1. Strong links:** The C_{sd} between the source s and destination d is high, and hence they are less prone to packet collisions from subtle interference. 1-2, 3-4, 5-6 are examples of strong links at all transmission powers.
- 2. Gray-zone links:** The C_{ij} for these links are not high, and these links have high Packet Error Rate (PER) [27]. Link 2-5 and 2-3 are gray-zone links at transmission powers less than 3 dBm and 0 dBm, respectively.
- 3. Unconnected links:** The C_{ij} for these links cannot be estimated by the measurement protocol since packets could not be transferred over these links – even at the maximum transmission power of 15 dBm. Hence, we assume that these node-pairs do not cause interference to each other. This assumption leads to incorrect results under some circumstances as we explain later. Links 1-6, 1-5, 3-5, 3-6, 4-5 and 4-6 are unconnected links.

We conjecture that the effectiveness of the CLP is based on the types of the links involved and whether the C_{ij} for all

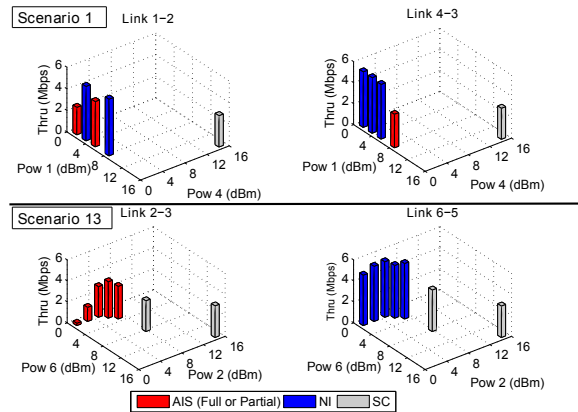


Fig. 14. Throughput at different tx power combinations

node pairs can be calculated.

Case 1: CLP optimizes the interactions when: (1) the link between the source and destination are strong, and (2) link between $S1-D2$ and $S2-D1$ either a strong or gray-zone links. This is because that the CLP has an accurate estimate of the interference caused by one source at the destination of the other link. Hence, CLP is successful in mitigating bad interactions in scenarios 1 and 3 in Table I.

We illustrate on the effectiveness of the CLP parameter selection by Figure 14. The top part of the figure depicts the throughput observed in Scenario 1 for different values of transmission power at the sources. NI, SC and AIS interactions are marked with blue, gray and red bars, respectively.

In the default setting, both sources transmit at 15 dBm, and hence the links have SC interaction. Both the links achieve around 3 Mbps. The optimal value of transmit power occurs when both nodes 1 and 4 transmits at 3 dBm where NI interaction is observed between the links. If the power varies beyond this region, then the links end up packet collisions. For example, when nodes 1 and 4 transmit at 8 dBm and 1 dBm, respectively, then link 4-3 suffers from partial packet collisions due to weaker received signal strength (Partial AIS). However, link 4-3 achieves full throughput when node 4 reduces the power to 5 dBm. But, link 1-2 now suffers from collisions because of the low received signal strength. The optimal point is found when 1 and 4 transmit at 3 dBm. If node 1 lowers the

power further, link 1-2 becomes vulnerable to collisions due to low signal strength.

Case 2: CLP also optimizes two strong links when s_1-d_2 and s_2-d_1 are unconnected and spatially far away. Here the sources cause low interference to other destination, and hence, our assumption that these nodes do not interfere is fairly accurate (Scenarios 2,4,5).

Case 3: CLP is partially successful in reaching its aims when at least one source and the other destination are unconnected links, and if the distance between these nodes are smaller. This phenomenon occurs in Scenarios 6 to 9, where nodes 1-5, 1-6, 2-5 and 2-6 are unconnected, but we believe that the receiver would experience non-negligible interference from the other source.

Case 4: CLP results in completely unoptimized interactions between two links when: (1) one of the link is in gray-zone, and (2) the receiver of one link is unconnected to the other source, but the nodes have non-negligible interference (similar to case 3). Scenarios 10-13 suffer from detrimental interactions because of the above reason (link 2 – 3 is a gray-zone link).

We illustrate Case 4 with a scenario 13, where link 2-3 suffers from complete packet collision. The bottom part of Figure 14 shows the possible throughput observed when the sources transmit at different powers. In the standard setting of 15 dBm transmit power, both links can sense each other, and hence observe SC interaction. As we lower the power of the source 2 below 5 dBm, link 2-3 suffers from lesser throughput due to constant interference from node 6. CLP fails to account for interference from 6, since nodes 2 and 6 are measured to be non-interfering.

Discussion on Deployment Challenges: We have assumed that CS threshold (c_i) is an energy threshold that can be set in the transceiver. In reality, this assumption is complicated by two factors: (1) changes to noise or received signal strengths over time, when we assume constant values, and (2) transceivers may use different logic to detect that a channel is busy, and we assume that CS Threshold is a simple energy threshold that can be set. We believe that the model can be extended with simple heuristics to provide effective interactions in such cases as we elaborate below.

First, our ILP model provides a feasibility range for carrier sensing c_i . Hence, the actual carrier sensing threshold can be set to a value which lower than the maximum of feasible c_i to absorb the effect variations of received signal strength or noise floor, and to avoid setting a threshold that is significantly close to the noise floor.

Second, our model can provide useful hints to transceivers employing different mechanisms for CS. Our model can be directly applied to some transceivers where carrier sensing is based on a single energy threshold (like CC2420 transceiver, which used in micaZ and Telos) [37]. Other transceivers compute instantaneous noise floor, and then conclude that the channel is busy if the difference between the signal and noise floor is less than some threshold T [8]. In such transceivers, nodes can exchange the observed noise in addition to the measured RSSI values, and adjust the threshold T accordingly.

Some transceivers, including Artheros transceivers [38], use other carrier sensing estimation algorithms such as mea-

suring the minimum signal strength needed to decode the preamble [8]. We believe that the range of c_i can be used in conjunction with the existing adaptive mechanisms. Our current experiments using Atheros cards indicate that our scheme works for links that have good signal strength; these links are more important to the higher level protocols, such as routing, and applications.

VII. CONCLUSIONS AND FUTURE WORK

We proposed a framework to optimize CSMA wireless networks by configuring different transceiver parameters to provide greater spatial reuse, while eliminating hidden terminals. We proposed an optimal model for two-link topologies, and an approximate algorithm for general topologies. We developed a distributed protocol based on the general algorithm that adapts dynamically to changes in the network. We evaluated the model, algorithms and protocol through simulations and testbed studies. The protocol was able to almost completely eliminate destructive interactions resulting in large improvements in throughput and delay.

We plan to extend this work to improve both the base algorithm and the protocol. We plan to formulate model and design near-optimal protocols that account for multi-rate multi-channel networks. Since existing wireless cards limit the number of parameters that can be measured or altered we plan to use Software Defined Radios (SDR) to provide greater flexibility in controlling PHY and MAC parameters.

Acknowledgements: This work was partially supported by National Science Foundation grants CNS-0958501, CNS-0751161, and Qatar National Research Fund grant number NPRP 08-562-1-095.

REFERENCES

- [1] The IEEE Working Group, "IEEE 802.11 Wireless Local Area Networks," 2002.
- [2] F. A. Tobagi and L. Kleinrock, "Packet Switching in Radio Channels: Part II—The Hidden Terminal Problem in Carrier Sense Multiple-Access and the Busy-Tone Solution," *IEEE Trans. on Communications*, vol. 23, no. 12, pp. 1417–1433, 1975.
- [3] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: a media access protocol for wireless LAN's," *SIGCOMM Comput. Commun. Rev.*, vol. 24, no. 4, pp. 212–225, 1994.
- [4] M. Garetto, J. Shi, and E. W. Knightly, "Modeling media access in embedded two-flow topologies of multi-hop wireless networks," in *MobiCom '05*. New York, NY, USA: ACM Press, 2005, pp. 200–214.
- [5] S. Razak, V. Kolar, and N. B. Abu-Ghazaleh, "Modeling and analysis of two-flow interactions in wireless networks," *Ad Hoc Networks*, vol. 8, no. 6, pp. 564 – 581, 2010.
- [6] X. Yang and N. Vaidya, "On physical carrier sensing in wireless ad hoc networks," in *IEEE INFOCOM*, vol. 4, march 2005, pp. 2525 – 2535.
- [7] T.-S. Kim, H. Lim, and J. C. Hou, "Improving spatial reuse through tuning transmit power, carrier sense threshold, and data rate in multihop wireless networks," in *MobiCom '06*. NY: ACM, 2006, pp. 366–377.
- [8] K. Jamieson, B. Hull, A. Miu, and H. Balakrishnan, "Understanding the real-world performance of carrier sense," in *ACM SIGCOMM workshop, E-WIND '05*. New York, NY, USA: ACM, 2005, pp. 52–57.
- [9] M. Vutukuru, K. Jamieson, and H. Balakrishnan, "Harnessing exposed terminals in wireless networks," in *USENIX NSDI'08*. Berkeley, CA, USA: USENIX Association, 2008, pp. 59–72.
- [10] Y. Gao, J. Hou, and H. Nguyen, "Topology control for maintaining network connectivity and maximizing network capacity under the physical model," in *IEEE INFOCOM 2008*, 2008.

- [11] V. Kolar, S. Razak, and N. B. Abu-Ghazaleh, "Interaction Engineering: Taming of the CSMA," in *ACM MSWIM '10*. New York, NY, USA: ACM, 2010, pp. 160–169.
- [12] A. Muqattash and M. Krunz, "A single-channel solution for transmission power control in wireless ad hoc networks," in *ACM MobiHoc '04*. New York, NY, USA: ACM, 2004, pp. 210–221.
- [13] T.-Y. Lin and J. Hou, "Interplay of spatial reuse and sinr-determined data rates in csma/ca-based, multi-hop, multi-rate wireless networks," in *IEEE INFOCOM 2007*, may 2007, pp. 803–811.
- [14] K. Whitehouse, A. Woo, F. Jiang, J. Polastre, and D. Culler, "Exploiting the capture effect for collision detection and recovery," in *IEEE EmNets '05*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 45–52.
- [15] V. Mhatre, K. Papagiannaki, and F. Baccelli, "Interference Mitigation Through Power Control in High Density 802.11 WLANs," in *IEEE INFOCOM 2007*, may 2007, pp. 535–543.
- [16] I. Broustis, K. Papagiannaki, S. Krishnamurthy, M. Faloutsos, and V. Mhatre, "Measurement-Driven Guidelines for 802.11 WLAN Design," *Transactions on Networking*, vol. 18, no. 3, pp. 722–735, 2010.
- [17] D. Halperin, T. Anderson, and D. Wetherall, "Taking the sting out of carrier sense: interference cancellation for wireless LANs," in *ACM MobiCom '08*. New York, NY, USA: ACM, 2008, pp. 339–350.
- [18] M. Z. Brodsky and R. T. Morris, "In defense of wireless carrier sense," in *ACM SIGCOMM '09*. New York: ACM, 2009, pp. 147–158.
- [19] H. Ma, R. Vijayakumar, S. Roy, and J. Zhu, "Optimizing 802.11 wireless mesh networks based on physical carrier sensing," *Networking, IEEE/ACM Transactions on*, vol. 17, no. 5, pp. 1550–1563, oct. 2009.
- [20] K.-P. Shih, Y.-D. Chen, and C.-C. Chang, "A physical/virtual carrier-sense-based power control mac protocol for collision avoidance in wireless ad hoc networks," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 22, no. 2, pp. 193–207, feb. 2011.
- [21] L. Jiang and J. Walrand, "Approaching Throughput-Optimality in Distributed CSMA Scheduling Algorithms With Collisions," *Networking, IEEE/ACM Transactions on*, vol. 19, no. 3, pp. 816–829, june 2011.
- [22] Z. Abichar and J. Chang, "A medium access control scheme for wireless lans with constant-time contention," *Mobile Computing, IEEE Transactions on*, vol. 10, no. 2, pp. 191–204, feb. 2011.
- [23] 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 *SIGCOMM '05*, 2005, pp. 121–132.
- [24] V. Kolar, K. Bharath, N. B. Abu-Ghazaleh, and J. Riihijrvi, "The effect of contention in CSMA networks: Model and fairness protocol," *Performance Evaluation*, vol. 68, no. 9, pp. 782–805, 2011.
- [25] B. Nardelli, J. Lee, K. Lee, Y. Yi, S. Chong, E. Knightly, and M. Chiang, "Experimental evaluation of optimal csma," in *IEEE INFOCOM*, april 2011, pp. 1188–1196.
- [26] A. Aziz, D. Starobinski, and P. Thiran, "Understanding and tackling the root causes of instability in wireless mesh networks," *IEEE/ACM Trans. Netw.*, vol. 19, pp. 1178–1193, August 2011.
- [27] V. Kolar, S. Razak, P. Mahonen, and N. B. Abu-Ghazaleh, "Link quality analysis and measurement in wireless mesh networks," *Ad Hoc Networks*, vol. 9, no. 8, pp. 1430–1447, 2011.
- [28] J. Padhye, S. Agarwal, V. N. Padmanabhan, and Lili, "Estimation of Link Interference in Static Multi-hop Wireless Networks," in *Internet Measurement Conference*, 2005.
- [29] J. Lee, S.-J. Lee, W. Kim, D. Jo, T. Kwon, and Y. Choi, "Rss-based carrier sensing and interference estimation in 802.11 wireless networks," in *IEEE SECON '07*, june 2007, pp. 491–500.
- [30] "Qualnet network simulator," <http://www.scalable-networks.com/>.
- [31] N. Ahmed, U. Ismail, S. Keshav, and K. Papagiannaki, "Online estimation of rf interference," in *ACM CoNEXT*. New York: ACM, 2008, pp. 4:1–4:12.
- [32] J. Yeo, "CRAWDAD trace dartmouth/campus/syslog/aplocations_2008 (v. 2009-09-09)," Downloaded from http://crawdad.cs.dartmouth.edu/dartmouth/campus/syslog/aplocations_2008, Sep. 2009.
- [33] R. Jain, D.-M. Chiu, and W. Hawe., "A quantitative measure of fairness and discrimination for resource allocation in shared computer system." *Technical Report 301, Digital Equipment Corporation*, 1984.
- [34] "Ffmpeg and libavcodec." [Online]. Available: <http://ffmpeg.org/>
- [35] "Xiph test media." [Online]. Available: <http://media.xiph.org/video/derf/>
- [36] "Madwifi: Linux driver for atheros cards," <http://madwifi.org/>.
- [37] K. Srinivasan and P. Levis, "RSSI Is Under Appreciated," in *IEEE Third Workshop on Embedded Networked Sensors (EmNets)*, 2006.
- [38] L. Scalia, I. Tinnirello, and D. Giustiniano, "Side Effects of Ambient Noise Immunity Techniques on Outdoor IEEE 802.11 Deployments," in *IEEE GLOBECOM*, 2008, pp. 1–6.



Vinay Kolar works at IBM Research, India. Previously, he worked at Carnegie Mellon University, Qatar and RWTH-Aachen University, Germany. He received his Ph.D. in Computer Science from State University of New York at Binghamton. His research interests are in the analysis and design of reliable and efficient networking systems.



Saquib Razak is an assistant professor at Carnegie Mellon University in Qatar. He completed his Ph.D. in Computer Science from State University of New York, Binghamton. He has a BS and MS degree in Electrical Engineering from University of Texas, Austin. His research areas include wireless networks and computer science education.



Nael B. Abu-Ghazaleh is an Associate Professor in the Department of Computer Science at the State University of New York at Binghamton. He received his Ph.D. and MS degrees in Computer Engineering from the University of Cincinnati. His research interests are in ad hoc networks, sensor networks, security and parallel computing.