



Measurement-driven admission control on wireless backhaul networks

Irfan Sheriff*, Prashanth Aravinda Kumar Acharya, Elizabeth M. Belding

Department of Computer Science, University of California, Santa Barbara, CA 93106, USA

Abstract

IEEE 802.11 wireless networks perform poorly in the presence of large traffic volumes. Measurements have shown that packet collisions and interference can lead to degraded performance to the extent that users experience unacceptably low throughput, which can ultimately lead to complete network breakdown [12]. An admission control framework that limits network flows can prevent network breakdown and improve the performance of throughput and delay-sensitive multimedia applications. In this paper, we present a measurement-driven admission control scheme that leverages wireless characteristics for intelligent flow control in a static wireless network. Experiments on the 25 node UCSB MeshNet show that the proposed admission control scheme can enhance network performance such that the QoS requirements of real time applications, such as VoIP, can be met.

© 2008 Published by Elsevier B.V.

Keywords: Wireless mesh networks; Admission control

1. Introduction

The deployment and usage of IEEE 802.11 wireless networks for Internet access has increased manyfold in recent years. According to a recent report, the usage of WiFi service (from a single provider) increased by 111% in the short time-span of 10 months [26]. Several cities around the world have announced plans to deploy (or have already deployed) city-wide 802.11-based networks that provide free Internet connectivity. These large networks offer use to thousands of users simultaneously. If the growth in the usage of wireless networks continues along current trends, these networks will soon become overutilized and congested. Unsatisfactory user experiences in city-wide networks have already led to questions about the ability of 802.11-based networks to sustain large traffic volumes [31]. With the growing usage of wireless networks and the increasing bandwidth requirements of current applications, these networks will suffer from increased levels of conges-

tion and eventually breakdown. To study congestion in currently deployed networks, Jardosh et al. present two case studies of operational 802.11 WLANs that experienced network breakdown [12]. These networks, deployed at Internet Engineering Task Force (IETF) meetings, consisted of over 100 access points (APs) with more than 1000 simultaneous users. Measurements showed that frequent packet collisions and interference led to degraded network performance to the extent that users experienced unacceptably low throughput and, in many cases, failed to maintain an association with any AP. The result was sparse or no connectivity for all the users in the network and an eventual network breakdown. The fundamental cause of the problem of congestion and degraded performance is the shared nature of the wireless medium. This problem can also be attributed to the design of the IEEE 802.11 protocol. Nodes in an 802.11-based network contend for access to the same spectrum. This is unlike other wireless networks, such as cellular networks, where the bandwidth required for a flow is reserved during the call-setup phase. Also, the IEEE 802.11 DCF mode employs CSMA/CA-based channel access, which creates the classic hidden terminal and exposed terminal problems. These problems can have a detrimental effect on the network performance.

* Corresponding author. Tel.: +1 805 450 0455; fax: +1 805 893 8553.

E-mail addresses: isherrif@cs.ucsb.edu (I. Sheriff), acharya@cs.ucsb.edu (P.A.K. Acharya), ebelding@cs.ucsb.edu (E.M. Belding).

Admission control solutions that limit the traffic in 802.11-based wireless networks to sustainable levels can prevent situations of network congestion. Although there has been significant research on admission control for 802.11-based wireless networks, there lacks a realistic system implementation that can limit traffic in these networks. This gap between proposed solutions and their actual deployment is, in many cases, due to the existence of unrealistic assumptions that render a system implementation infeasible. Other admission control solutions require modifications to the hardware. These solutions are not suited for networks that are already deployed. Therefore, there exists a need for an admission control solution that both takes into account the behavior of a real-world 802.11 network and is implementable on commodity off-the-shelf radios.

In this paper, we present measurement-driven admission control (MDAC), a measurement-driven framework for admission control in wireless networks. The framework uses network measurements to characterize the behavior of the wireless channel and continuously measures the availability of resources in the network. The resource availability is measured in the form of the time fraction for which the wireless medium is free. This information is then used in the decision-making process that determines whether to admit new flows into the network.

Our contributions in this paper are twofold. First, we present an analysis of wireless network behavior that is essential to perform admission control. Second, we present the design, implementation and evaluation of an admission control scheme that can intelligently limit the flows in the network, thereby preventing congestion, while at the same time achieving efficient utilization of the wireless medium. The remainder of the paper is organized as follows. Section 2 presents a simple experiment that illustrates the need for an admission control solution. Section 3 lists the assumptions and terminology used in the paper. We present our findings about the characteristics of wireless links in Section 4. Section 5 describes the design of the admission control scheme. We present the details of our implementation and the results from evaluation in Sections 6 and 7, respectively. Section 8 discusses some of the issues and challenges of our scheme. In Section 9 we contrast our work with existing literature and, finally, we conclude in Section 10.

2. Motivational scenario

We perform a simple experiment to understand the extent of deterioration in network performance in the absence of admission control. The experiment also demonstrates the need for an admission control scheme in wireless networks. We first describe the experimental testbed.

2.1. Testbed description

All the experiments described in this paper were conducted in the UCSB MeshNet, an indoor wireless testbed

which consists of 25 wireless nodes [30]. All nodes in the testbed use 802.11a/b/g cards based on the Atheros chipset. Several nodes in the testbed have multiple radios. However, in this paper we only use one radio of each node, operating in the 802.11b/g mode. Each node is also equipped with an Ethernet interface that is used to control the node during experiments, thus ensuring that the experiment control traffic does not affect the wireless network experiments. The nodes use Linux (kernel version 2.4) as their operating system. We use the open source MadWifi [28] driver v0.9.2 to control the cards. RTS/CTS is disabled for all the radios. The nodes are placed in different locations on three floors of the building. The testbed coexists with an 802.11b wireless LAN that provides Internet connectivity throughout the building.

2.2. Orchestrating congestion

We create a scenario on the UCSB MeshNet testbed to understand the extent of damage caused by uncontrolled flow admission in the network. We consider the flows in the network to belong to real-time applications such as voice over IP (VoIP), i.e., they are delay-sensitive and have strict throughput requirements. We initiate 15 64 kbps CBR flows that imitate VoIP calls between random node pairs who are in the immediate neighborhood of each other. Each flow lasts for 300 s and the flow arrival rate is one every 10 s. Fig. 1a and b shows the throughput and delay performance of the flows over time.

We make two observations from these graphs. First, the impact on throughput and delay is drastic and we can clearly demarcate the time beyond which the network starts to collapse; the 150 s time on the graphs represents this point. Second, congestion can spread and affect the flow behavior across the entire network. In the scenario shown, the throughput and delay of the flows in the first 150 s of the experiment are within the limits acceptable to the VoIP application. When the twelfth flow is admitted in the network at around 150 s into the experiment, the throughput and delay performance of most of the flows begin to rapidly deteriorate and the ongoing flows no longer receive their needed QoS. It is thus clear that congestion at one point in the network could cause detrimental effects across the entire network.

Therefore, unrestricted flow admission in the network affects the quality of service available to new flows as well as established flows. This observation motivates our design of an admission control scheme to perform intelligent flow control in the network.

3. Assumptions and terminology

In this section, we first describe the attributes of the networks for which we design our admission control scheme. Then, we define some of the common terms used throughout the paper.

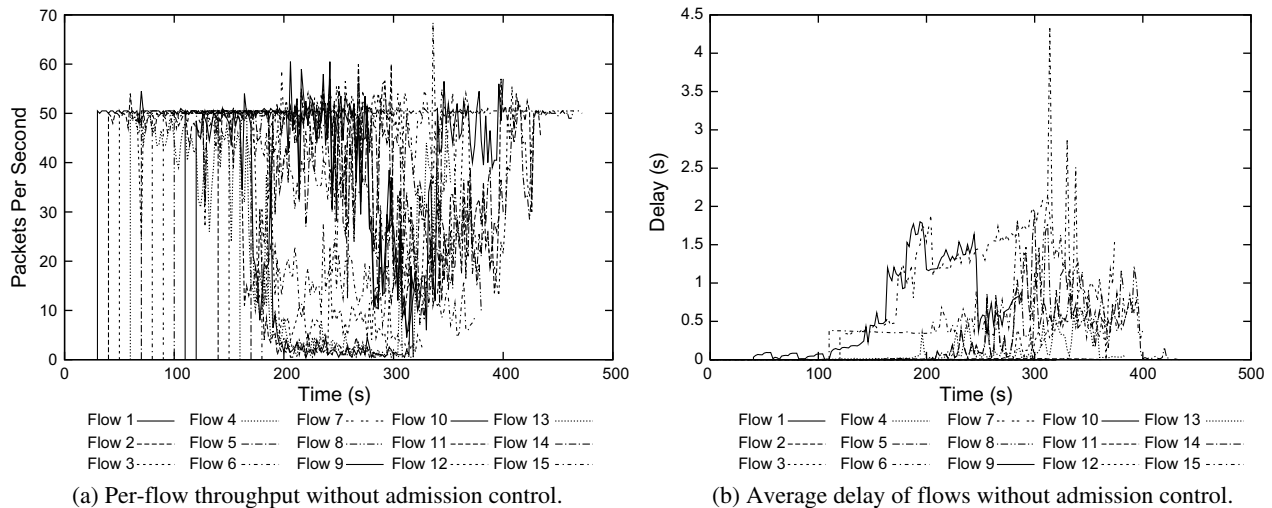


Fig. 1. Performance when flow admission is unrestricted on the UCSB MeshNet.

165 3.1. Network attributes

166 Our admission control scheme is designed with certain
 167 network attributes in mind. For instance, we design our
 168 admission control solution for the generic setting of static
 169 multi-hop wireless networks such as backhaul mesh net-
 170 works and ad hoc networks with stationary nodes. As a
 171 first step in providing a practical admission control solu-
 172 tion for such networks, we focus on the problem of admis-
 173 sion control in a simplified network scenario that consists
 174 of traffic between single-hop neighbors only. However,
 175 much of the discussion in the paper is generic to networks
 176 with multi-hop flows. We believe that a simple extension of
 177 our scheme, as discussed in Section 8, can support multi-
 178 hop flows. We intend to describe this extension as part of
 179 future work. Our current solution is also applicable to
 180 802.11 wireless LANs, but may be of limited utility in the
 181 presence of highly mobile client nodes.

182 In the design of our admission control scheme, we focus
 183 specifically on the support of networks in which the aver-
 184 age traffic load, in terms of packet rate and packet size, is
 185 predictable. This is a characteristic of, for instance, VoIP
 186 traffic. The success of VoIP over the wireline Internet leads
 187 us to expect that this application will be key in continuing
 188 to drive the wide acceptance and commercial success of
 189 wireless networks. A broad range of wireless VoIP solu-
 190 tions are commercially available and are being deployed
 191 in enterprises and campuses [10,25]. Because of the incipi-
 192 ent widespread adoption of these and similar devices, we
 193 focus our admission control scheme on traffic that exhibits
 194 predictable performance. However, we discuss the perfor-
 195 mance of our scheme in a network that carries a mix of
 196 real-time and best-effort traffic in Section 8.

197 3.2. Terminology

198 In this paper, we use the term ‘interference’ to refer to
 199 the impact of concurrent transmission on packet recep-

200 tions. An interfering neighbor thus refers to a node that
 201 can affect the successful reception of packet transmissions
 202 at another node. We use the term ‘carrier sensing’ to refer
 203 to sensing the medium to assess whether it is free. The term
 204 ‘busy-time’ is used to refer to the time duration for which
 205 the channel is not free. A carrier sensing neighbor is thus
 206 a node that causes busy-time at a given node X and, there-
 207 fore, can cause deferral of the data transmissions at X . The
 208 term received signal strength indicator (RSSI) refers to the
 209 signal strength of a packet (in dBm) relative to the noise
 210 floor at the radio. The default value of the noise floor for
 211 Atheros radios is -95 dBm. Note that the above definition
 212 of RSSI is specific to Atheros radios. The term received sig-
 213 nal strength (RSS) refers to the absolute energy level of a
 214 packet and is measured in dBm. The RSS value of a packet
 215 can be computed using the RSSI value reported by the
 216 Atheros radio and the noise floor reported by the radio.

4. Understanding wireless characteristics

217 Many previously proposed admission control solutions
 218 are based on a common set of assumptions that may not
 219 hold true in real-world 802.11 networks. In the following
 220 sections, we evaluate the accuracy of some of the well-
 221 known assumptions used by these existing solutions. In
 222 particular, we investigate the medium time consumption
 223 at neighbor nodes; the possibility of communication
 224 between carrier sense neighbors; and the behavior of
 225 packet receptions, interference and temporal variations in
 226 link quality in wireless networks. We then incorporate
 227 the insights obtained from these studies in the design of
 228 our admission control scheme described in Section 5.
 229

4.1. Impact on neighbor busy-time

230 A common assumption of many existing admission con-
 231 trol techniques is that the impact of a new flow on the busy-
 232 time of the surrounding nodes is binary. In other words, in
 233

the simplest case, the transmission of a packet by a node has no effect on the busy-time at nodes outside the transmitter's carrier sense range, while the increase in busy-time at nodes within the transmitter's carrier sense range is equal to the duration of the packet transmission. Hence, when a new flow is admitted, the increase in busy-time at the carrier sense neighbors is equal to the transmission time of the new flow. To investigate whether this is indeed the case, we perform the following set of experiments. Our experiments are performed on the UCSB MeshNet, and are conducted during the night to minimize the effect of interference from co-located wireless networks.

We use the reverse-engineered Open HAL [29] implementation of the MadWifi driver for the Atheros AR5212 chipset radios to understand the relationship between busy-time and communication range. Atheros maintains register counters to track the 'medium busy time' and the 'cycle time'. The cycle time counter is incremented at every clock tick of the radio and the medium busy counter represents the number of clock ticks for which the medium was sensed busy. The ratio of these two counters thus represents the medium busy-time fraction. Note that the medium busy-time fraction includes the time spent by the radio for transmission and reception of packets.

In our experiment, a receiver node is placed at a fixed location in the network. A sender node is placed at different locations in the network to vary the packet reception rate at the receiver. Keeping the receiver stationary ensures that the variation in the environmental noise at the receiver is minimal. The sender transmits fifty 100 byte broadcast packets per second at 1 Mbps. The receiver tracks the packet reception rate and also estimates the medium busy-time fraction caused by the sender. The medium busy-time is estimated as the difference in the medium busy-time values reported by the radio before and after the experiment. To obtain the medium busy-time fraction, the measured value of busy-time is normalized with respect to the difference in the reported medium busy-time when the sender is close to the receiver and 100% of the packets are received successfully. The experiment lasts 30 s at each location of the sender.

Fig. 2 shows a plot of the calculated busy-time fraction at the receiver versus the fraction of packets received by the receiver for different locations of the sender node. The graph shows that medium busy-time fraction of a node on its neighbor ranges anywhere from 0 to 1 and is not a binary relationship. For example, when 40% of the transmitted packets are received, the busy-time fraction varies from 0.4 to 0.5.

The experimental results imply that, in a larger network, the bandwidth consumption of a flow on each carrier sense neighbor is not 100% of the bandwidth requirement of the flow. On the contrary, the impact is likely to vary based on the location of the neighbor, as well as other environmental factors. Thus, existing admission controls schemes that assume a new flow will consume bandwidth equivalent to the data rate of the sender can be highly pessimistic and

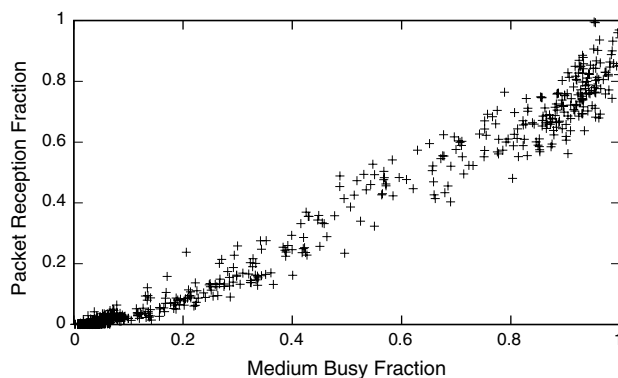


Fig. 2. Packet reception at 1 Mbps data rate versus medium busy fraction.

are likely to overestimate the resource requirements at carrier sense neighbors. We incorporate the above result into our admission control scheme by computing the fractional increase in busy-time relative to the transmission rate of the sender at the carrier sensing neighbors of the transmitting node. The computation procedure is described in Section 5.

4.2. Communication with carrier sensing neighbors

Many existing admission control techniques require sender nodes to communicate with nodes outside of their reception range, but within their carrier sensing range. These studies often assume the carrier sense range is about twice the transmission range [23,9,21] and, therefore, direct communication with carrier sense neighbors is not possible. Some techniques use high power transmissions or multi-hop forwarding to communicate with carrier sense neighbors [23]; others propose techniques to adjust a node's carrier sense range so that it can hear the transmissions of potentially interfering neighbors [11]. However, recent test-bed measurements show that the simple assumption that the carrier sense range is twice the transmission range does not hold in real-world networks [16].

In Section 4.1, we showed that the increase in busy-time caused by a sender on its neighbor can be any range from zero to 100% of the actual transmission time. This observation implies that the sender causes its neighbor to defer packet transmissions for some fraction of the transmission time. We are interested in understanding whether nodes that cause transmission deferral at each other can communicate, and if so, the extent of communication achievable between these carrier sensing neighbors. We again refer to Fig. 2 to understand the possibility of communication between carrier sensing neighbors. The graph shows that the receiver node is able to receive a non-zero number of packets from a transmitting neighbor that induces more than 5% of medium busy fraction. The probability of receiving a packet from a neighbor increases with the increase in busy-time the neighbor induces at the receiver node. For example, a node can receive about one out of five packets from a neighbor that induces 40% busy-time and about three out of five packets from a neighbor that

induces 80% busy-time. The same experiment repeated at a higher data rate would result in much lower packet reception rates for similar values of busy-time fraction. Additionally, the reception capability of a radio depends on its sensitivity, thermal noise level and other environmental factors. However, multiple experiments with different off-the-shelf radios showed that the packet reception at 1 Mbps is achievable at low busy-time values for a majority of the radios.

To understand the reason for successful packet reception inside carrier sense range, we refer to the datasheet for the Atheros radios. The data sheet states that a packet sent at 1 Mbps can be successfully received when the RSSI is as low as zero. In contrast, the Clear Channel Assessment (CCA) threshold for Atheros radios is reported to be about -81 dBm, which corresponds to an RSSI of 14 (assuming the noise floor to be the default -95 dBm) [17]. The CCA threshold is used to determine whether the channel is busy. The significant difference between the CCA threshold and the minimum RSSI required for packet reception at 1 Mbps indicates that nodes can communicate with carrier sensing neighbors using low data rate packets. Prism radios were found to have CCA values as high as 40 dB above the noise floor [27]. Based on these results, we have shown that a node can communicate with all the carrier sensing neighbors that have non-negligible impact on the busy-time of the node. The different values of packet reception indicate that this communication may be unreliable. In our admission control scheme, we propose to use this unreliable reception of low data rate packets to periodically communicate the resource availability information of a node to its carrier sensing neighbors. Resource estimation at carrier sense neighbors is essential for admission control decisions. We add sufficient redundancy in the communication to ensure that, with a high probability, all the carrier sense neighbors that have non-negligible impact on the busy-time receive resource availability information.

4.3. The RSS–Packet Reception relationship

We now study the relationship between the packet reception rate and the average RSS of the received packets at different data rates. Reis et al. [17] study the behavior of wireless links in static networks and show that the packet reception probability is a function of the RSS of the received packets, and this relationship is specific to each node in the network. They propose a measurement-based model to characterize the packet reception rates of links in the network. The proposed model functions as follows: Each node records the RSS for the broadcast packets received from each of its neighbors during the network profiling experiments. These RSS values are then used to derive a mathematical function in the form of a piecewise linear curve that models the reception probability of the link at different RSS values. This curve can be used to predict the packet delivery probability for any RSS value.

This technique of predicting packet reception rate measures and predicts link quality using the RSS values of packets sent at the broadcast rate. Several other protocols [7,6,8] also rely on measurements based on broadcast packets to predict link quality at different rates. However, the SINR (signal to interference-noise ratio) requirements for packet reception are different for each of the supported 802.11 data rates. For example, the SINR required for packet reception at the 54 Mbps data rate is much higher (≈ 24 dB) than that at 6 Mbps (≈ 6 dB) [22]. In the absence of transmit power control, the radios transmit all packets at the same power level irrespective of the data rate of the packet. Therefore, link quality predictions based solely on measurements that use broadcast packets may suffer from significant inaccuracies.

The naïve approach to rectify this problem is to send probe packets at all possible rates. This approach, however, is highly inefficient and causes excessive load on the network. An alternative approach is to use the existing unicast traffic in the network to measure the link quality at different rates. To this effect, we note that the rate selection algorithm SampleRate [3] maintains statistics about the average number of packet retransmissions at different rates between each neighbor pair. Thus, the traffic utilized by SampleRate is an ideal candidate on which to piggyback the link quality measurements at different rates. An advantage of integrating the link quality measurement with SampleRate is that SampleRate maintains up-to-date statistics on all the rates that a node uses or is likely to use. Also, SampleRate does not send frequent probe packets for data rates that it is not likely to use. Therefore, this integration provides a scalable and feasible method of link quality estimation at different data rates.

We modify the MadWifi driver to maintain statistics about the RSS of each received packet on a per-rate basis for each neighbor. Each node periodically exchanges its SampleRate statistics with its neighbors. Each node then computes the RSS–Packet Reception relationship for each rate.

We deploy the modified MadWifi driver throughout our testbed and each node obtains the RSS–Packet Reception piecewise linear curves for different data rates. Fig. 3 shows the RSS–Packet Reception at different rates for two representative nodes in the testbed. We observe from the graphs that each data rate exhibits significantly different characteristics for the same RSS value. The graphs also show the different RSS requirements for the 802.11b and 802.11g rates that result from the difference in the underlying physical layer technology. Another observation from the graph is that there does not exist an obvious correlation between the curves for different rates. Therefore, extrapolating link quality based only on broadcast packets leads to errors in prediction. Note that some nodes in the network do not have packet reception data for all possible rates (e.g., Fig. 3(a)). This gap in the data is because SampleRate measures link quality only for those data rates that it is likely to use, which may be a subset of all the available data rates.

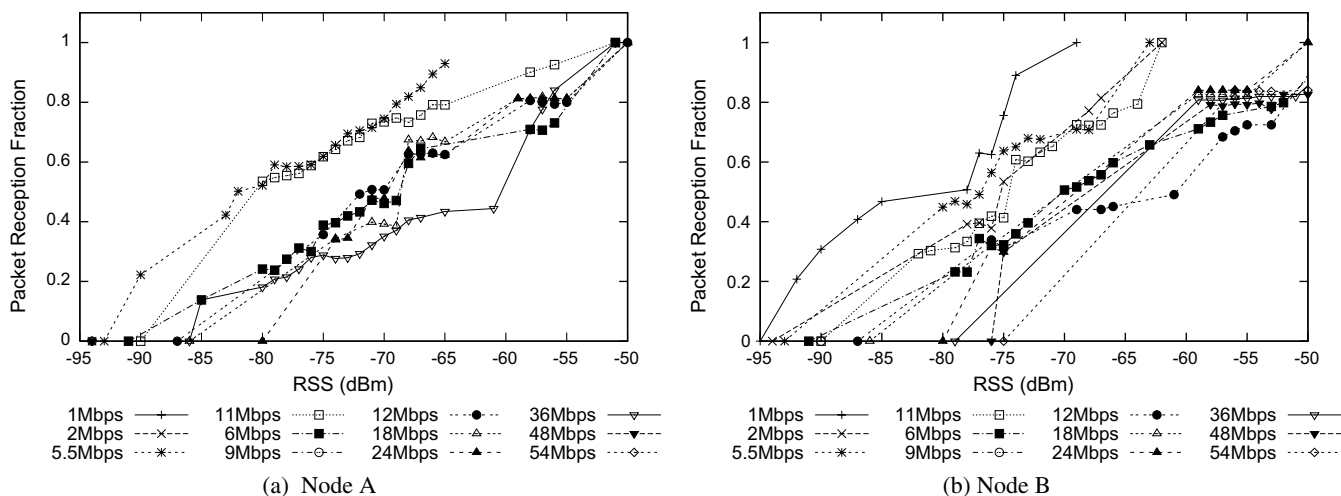


Fig. 3. Piecewise linear curves of the RSS–Packet Reception relationship for all data rates at two representative nodes.

Because these data rates are not likely to be used for transmission, the absence of this data does not affect the link quality predictions.

From the above result we learn the importance of measuring the link quality at each data rate individually. The implementation of our admission control scheme uses the modified MadWifi driver. The RSS–Packet Reception curves obtained from the driver are then used to estimate the impact of a new flow on packet reception.

4.4. Collisions

None of the existing approaches for admission control consider the impact of collisions on packet receptions at a neighbor node. The previous discussion about the busy-time estimate only concerns the extent to which the transmissions could be impaired at a neighboring node. On the other hand, a new flow that is admitted into the network can cause collisions at neighboring nodes and hinder packet receptions. Hidden terminals further exacerbate the extent of collision losses. A hidden node that admits a new flow can cause collisions even at a node that has sufficient free medium time to accommodate the new flow. The use of RTS/CTS reduces collisions but is typically not used in most network scenarios due to the additional per packet overhead [18]. Thus, it is imperative to account for collisions while designing an admission control metric.

To study the impact of collisions in a hidden terminal scenario, we conduct the following experiment. We create a three node linear topology with the receiver node placed in between two sender nodes. Each sender broadcasts fifty 100 byte packets per second at the 1 Mbps data rate. The second sender is a hidden terminal to the first sender and the experiment is repeated with eight different hidden terminal topologies. The topologies are chosen such that the difference in the average RSS of the packets from the two senders at the receiver is varied. The receiver node tracks the packet reception rate

as well as the average RSS of the received packets. For each of the topologies, we plot the fraction of packets transmitted by the first sender that are received successfully at the receiver in two scenarios: in the first, the hidden terminal (the second transmitter) does not transmit packets; in the second, the hidden terminal transmits packets as just described. Fig. 4 presents this graph. The graph shows that the difference in average signal strength of the two senders impacts the extent of packet collisions and, in turn, the packet reception rate. The impact of collisions on the packet reception rate reduces as the signal strength of the first sender increases with respect to that of the hidden terminal. Note that the impact of packet collisions shown is specific to the data rate used.

In our admission control scheme, we account for the impact of collisions caused by the admission of a new flow. We use the difference in signal strength of two senders along with the RSS–Packet Reception graphs described in Section 4.3 to estimate the impact of collisions.

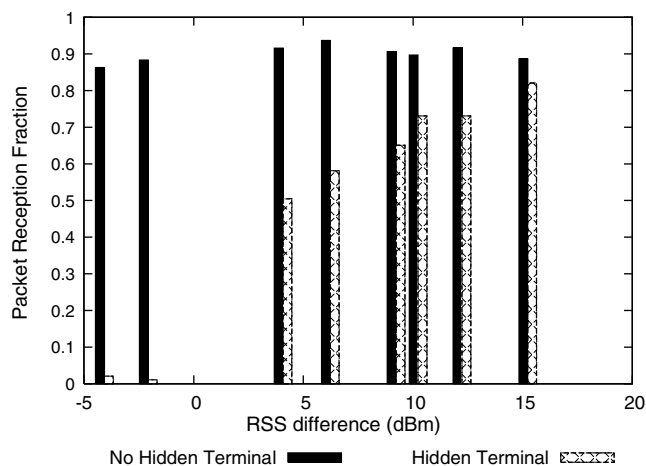


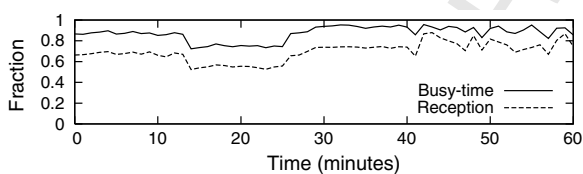
Fig. 4. Reduction in packet reception due to collisions.

499 4.5. Temporal behavior

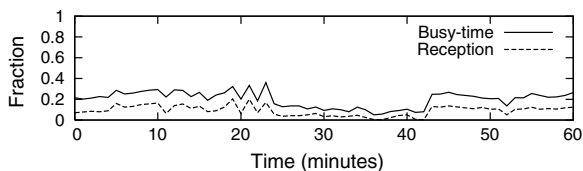
500 Previous work on measurement of packet receptions in
 501 static wireless networks has shown that, although packets
 502 losses occur in bursts, the quality of a majority of links
 503 in the network is stable when measured over long time
 504 intervals (on the order of minutes) [17,1]. This observation
 505 implies that the packet reception rate of a link measured
 506 during one time-window of suitable granularity can be used
 507 to accurately predict the packet reception probability dur-
 508 ing the subsequent time-window.

509 We explore the degree of stability of the busy-time met-
 510 ric and the appropriate time scale on which to measure and
 511 predict the busy-time metric. We conduct the following
 512 experiment on 50 randomly chosen links in the MeshNet
 513 testbed to study the temporal behavior of the busy-time
 514 metric of the links in the network. One end of the link acts
 515 as the sender node and the other as the receiver node. The
 516 sender node broadcasts fifty 100 byte packets per second at
 517 the 1 Mbps data rate. The receiver node records the pack-
 518 ets received from its sender and the medium busy-time frac-
 519 tion at the end of every 30-s interval. Each experiment lasts
 520 a duration of 60 min. Fig. 5 shows the variation of link
 521 qualities for three representative nodes in the testbed.

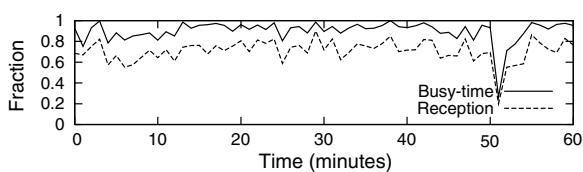
522 From the graphs, we first see that the packet reception
 523 rate exhibits the predictable behavior described in previous
 524 work. In addition, we see that the busy-time fraction has a
 525 high degree of correlation with the packet reception rate
 526 for all three links. This correlation exists for all three cate-
 527 gories of links shown in the graphs: stable links with high
 528 packet reception rate (Fig. 5(a)), stable links with low
 529 packet reception rate (Fig. 5(b)) and links that exhibit a
 530 higher degree of variability of packet reception rate



(a) Example link with a high packet reception rate.



(b) Example link with a low packet reception rate.



(c) Example link with variability in packet reception rate.

Fig. 5. Busy-time and reception behavior over a 1 h period for three representative links.

(Fig. 5(c)). The above observation on the correlation of
 531 busy-time fraction and packet reception rate illustrates that
 532 the busy-time metric exhibits predictability characteristics
 533 similar to the packet reception rate. In other words,
 534 although the impact on the busy-time of a neighbor node
 535 may vary rapidly in the short-term (milliseconds), it
 536 remains predictable over a longer period of measurement
 537 (tens of seconds). We compute Pearson's linear correlation
 538 coefficient to show the relationship between packet recep-
 539 tion rate and busy-time across all the chosen links at differ-
 540 ent data rates. We find that the two entities are highly
 541 correlated for all the links with 0.895 as the median value
 542 of the correlation coefficient. This implies that, similar to
 543 packet reception, the busy-time behavior of links can also
 544 be predicted for appropriate time scales. However, the pre-
 545 dictability of busy-time is affected by atypical events such
 546 as operation of a microwave.
 547

5. Admission control 548

549 The results from the experiments in the previous section
 550 can be summarized as follows. The impact of a node on the
 551 busy-time of its neighbor is non-binary; communication
 552 with neighbors that have a non-negligible impact on
 553 busy-time is achievable; packet reception at a node can
 554 be represented as a piecewise function of the RSS and data
 555 rate; the impact of overheard traffic on packet reception
 556 depends on the RSS of the packets, and the piecewise func-
 557 tion can be leveraged for predicting the packet loss; and the
 558 impact on busy-time is predictable with measurements over
 559 a period of seconds to minutes.

560 We build on these results and present the design of our
 561 measurement-driven admission control (MDAC) system
 562 that restricts flow admissions in a wireless network. The
 563 use of MDAC results in the reduction of network conges-
 564 tion that is likely to be created by the unrestricted admis-
 565 sion of flows in the network.

566 To perform admission control, we consider the band-
 567 width allocation, delay and jitter metrics. By controlling
 568 bandwidth allocation, delay and jitter can also be con-
 569 trolled [23]. Therefore, the metric of primary interest for
 570 our admission scheme is the available bandwidth. In a
 571 shared wireless medium, the two factors that determine
 572 the bandwidth available on a link are: (1) the fraction of
 573 time for which the medium is free and hence available for
 574 transmissions, and (2) the data rate used for transmissions.
 575 Each node has the transmission data rate information
 576 available locally. However, the fraction of time for which
 577 the medium is free must be computed.

578 Chakeres et al. show that metrics such as packet delay,
 579 average queue size, MAC layer contention window, and
 580 collisions provide incomplete information about the wire-
 581 less medium utilization [11]. Further, the authors show that
 582 the channel busy-time is a direct measure of the network
 583 utilization. Therefore, in our admission control scheme,
 584 we use the busy-time measurements to estimate the fraction
 585 of time for which medium is free.

In the following section, we present a discussion of 802.11 node behavior and interaction with neighbors. An understanding of the node interaction with neighbors is essential to identify the factors that affect the packet transmissions and receptions in the network upon the admission of a new flow.

5.1. Node behavior

An 802.11 wireless node can be in any one of the following states: Transmit (TX), Receive (RX), ChannelBusy and Free. We represent the fraction of time spent by a node in each of these states as t_{TX} , t_{RX} , t_{CB} and t_F , respectively. ChannelBusy represents the state during which a radio cannot attempt a packet transmission. Thus, t_{CB} includes the time during which the channel is busy (\bar{t}_{CCA}), and the time during which the 802.11 MAC spends additional wait time for the DIFS, SIFS and backoff periods (t_{MAC}). Note that $\bar{t}_{CCA} = t_{CCA} - t_{RX} - t_{TX}$, since t_{CCA} , the fraction of time the Clear Channel Assessment (CCA) indicates the channel to be busy, includes the transmission and receive time duration. In short, t_{CB} constitutes the unusable fraction of medium time that could be due to any of: channel noise, thermal noise, neighbor-to-neighbor packet transmissions or the MAC overhead. The remainder of the time constitutes t_F . The Free state time fraction, t_F , can be calculated as $t_F = 1 - (t_{CCA} + t_{MAC})$.

To understand the impact of a new flow in the network, we need to examine its effect on both the TX and RX states at a node. If the medium time consumption of the new flow is higher than the available time t_F at a node, the t_{TX} time at the node could be restricted. This behavior is due to the physical carrier sense mechanism of the IEEE 802.11 MAC protocol. Packets are sent into the medium only when medium is sensed free. The implication of this behavior is that packet transmissions can be throttled due to an increase in the ChannelBusy time, t_{CB} , caused by a new flow.

The impact of a new flow on the RX state depends on the signal strength of the current receptions. An increase in channel noise or neighbor packet transmissions could decrease the packet reception capability of the node. Packet losses during the RX state are likely to result in retransmissions from the sender, leading to an overall increase in t_{RX} at the node. With the use of rate selection algorithms, packet losses could have an additional impact of reducing the data rate used. This can further increase the t_{CB} at neighbor nodes and affect the performance of the flows in the network.

Thus, there are two factors that we consider important at every node during admission control: sufficient t_{TX} to allow packet transmissions as a sender, and sufficient tolerance to interference so as to limit the packet loss as a receiver.

In the remainder of the paper, we define Channelbusytime as the fraction of time spent in the TX, RX or ChannelBusy states. Therefore, Channelbusytime includes the

additional MAC overhead caused by the silence periods due to DIFS, SIFS and backoff. We can thus represent Channelbusytime as:

$$\text{Channelbusytime} = t_{CCA} + t_{MAC} \quad (1)$$

Note that throughout the paper, we use the term ‘busy-time’ to refer to t_{CCA} .

5.1.1. Example scenario

We now explain neighbor interaction with the help of an example scenario. Consider the example shown in Fig. 6, where a node pair $S-R$ needs to admit a new flow in the network. In our initial discussion, consider that the flow only consists of data packets in one direction from S to R ; we postpone the discussion on the impact of ACKs until later in this section. Assume that the flow requires t_{flow} fraction of the medium time. We examine the impact of the new flow on the different nodes in the network.

Once the new flow is admitted, t_{TX} at node S increases by t_{flow} . If there is sufficient free medium time at node S ($t_F > t_{flow}$), the impact on existing transmissions at S is negligible. However, without sufficient medium time ($t_{flow} > t_F$), packet transmissions at S could be throttled with the admission of the flow.

Consider node N to be a carrier sensing neighbor of node S . As a carrier sensing neighbor, the Channelbusytime at node N increases by up to t_{flow} with the admission of the new flow. The impact of this increase depends on the available medium time t_F at N . With sufficient t_F to accommodate the additional Channelbusytime consumption, the impact on existing transmissions from N is minimal. However, without sufficient t_F , the existing packet transmissions could be throttled at N .

Now, suppose that nodes A and B both have an existing flow to node N . The reception of packets at node N from senders A and B can be affected due to the collisions caused by the overheard traffic from node S . Note that node A is partially outside the carrier sense range of S and node B is completely outside the carrier sense range. Thus, nodes S and A share the medium and node B is a hidden terminal to node S . The rate of collisions is thus expected to be higher on transmissions from node B , assuming other fac-

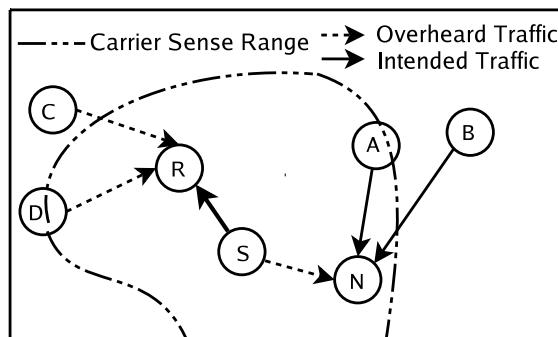


Fig. 6. Example of a node pair $S-R$ admitting a flow.

tors such as data rate, packet size and packet rate to be the same.

Packet collisions result in retransmissions from A and B . These retransmissions increase t_{RX} at node N and t_{CB} at other neighbors of A and B . This can cause a potential congestion *ripple effect* in the network. The reasoning is as follows. The typical response of a rate selection algorithm to packet loss is to reduce the data rate for more robust transmissions. When packets from nodes A and B experience collisions due to overheard traffic from node S , lower data rates are used. This leads to a further increase in t_{RX} at N and t_{CB} at other neighbor nodes of A and B . It also results in the packets from nodes A and B being more susceptible to overlap with other packets in the medium because the lower rate traffic consumes additional medium time. The impact of collisions can thus quickly spread throughout the network. Therefore, it is important to control the collisions at the neighbors of N to prevent such a congestion ripple effect.

Ensuring that the existing flows are not affected by the new flow does not suffice. Suppose nodes C and D are senders with existing flows to other nodes in the network (not shown in the figure). Node R is a neighbor of both nodes C and D and can overhear their transmissions. At node R , the new flow from S causes an increase in t_{RX} . This increase in t_{RX} makes the packets received at node R , including packets of the new flow, to be more susceptible to collisions caused by transmission from nodes C and D . In particular, the packets are more susceptible to collisions from hidden terminals such as node C . The above discussion about the impact of collisions at node N is applicable in this case also, with node R as the receiver. In this case, the overheard traffic is from the neighbor nodes C and D , and the intended traffic is from node S .

In summary, ensuring sufficient t_{TX} to allow neighbor packet transmissions and ensuring the loss imposed on neighbor packet receptions is minimal involves predicting the increase in Channelbusytime and the increase in packet collisions that can result with the admission of a new flow.

IEEE 802.11 requires transmission of an ACK packet for every data packet received. We consider the impact of the MAC layer ACKs from node R to S . The Channelbusytime consumption of ACK traffic is expected to be much less than that of data traffic. However, we cannot always ignore the impact of ACKs. For instance, if the data traffic consists of 1468 byte frames at 11 Mbps, then the Channelbusytime consumption of a 802.11b sender can be obtained as:

$$\begin{aligned} \text{Channelbusytime} &= DIFS + \text{preamble} + \text{data} \\ &= 50 + 192 + \frac{1468 * 8}{11 \text{ Mbps}} \\ &\approx 1300 \mu\text{s} \end{aligned}$$

In a similar manner, we can compute the ACK Channelbusytime to be about 300 μs . Thus, the ACK Channelbusytime consumption is only about 23% of the

Channelbusytime consumed by the data packets. However, with smaller data packets of about 160 bytes, as used in VoIP, the Channelbusytime of a single data frame at 11 Mbps is approximately 400 μs . The ACK Channelbusytime thus becomes comparable at smaller packet sizes of data traffic.

The packet collision impact of ACK packets does not depend on the carrier sense range of the node R . This is because ACKs are always sent after a SIFS wait upon reception of a data packet. This is in contrast to a data packet transmission that occurs after carrier sensing that the channel is free. This property of transmission of ACKs implies that R acts as a hidden terminal to nodes outside the carrier sense range of the node S . This is because neighbors of the node S that can overhear the data packet set the NAV value so as to not collide with the ACK packet.

With this understanding, we next present the design and operation of MDAC.

5.2. Operation

At a high-level, MDAC decides whether to admit a new flow into the network based on two considerations: first, the node and its neighbors have sufficient resources (medium time) to accommodate the new flow; and second, the admission of the new flow does not negatively affect the performance of existing flows. The two considerations of the scheme are evaluated with the help of five operational steps outlined below. Each of these steps is discussed in detail in the following sections.

1. Each node in the network maintains a profile for each of its neighbors. The neighbor profile includes statistics of average RSS, packet reception rates, number of packets received per second, busy-time, and link data rates. The neighbor profile is generated and maintained using either broadcast probes or the data traffic transmitted by the nodes. In addition to a neighbor profile, nodes also maintain statistics about the consumption of local resources, such as the medium Channelbusytime, the average number of packets sent and received per second, and the average data rate.
2. A node that wishes to admit a new flow estimates the availability of sufficient local bandwidth for packet transmissions. The available local bandwidth is predicted using the local medium Channelbusytime information.
3. The node then estimates whether the new flow will throttle on-going packet transmissions at any of its neighbors. The medium Channelbusytime information from the neighbor profile and the fractional Channelbusytime impact of the new flow is used to arrive at this decision.
4. Next, the node estimates whether the packet collisions caused by the new flow at any of the neighboring nodes exceed a certain threshold. The Channelbusytime and packet statistics from the neighbor profile of the two-hop neighborhood are leveraged for this decision.

5. Finally, the node estimates whether the packet collisions caused by the existing traffic on the new flow exceed a certain threshold. This decision is based on the medium Channelbusytime from the neighbor profile and packet statistics collected at the receiver node.

A new flow is admitted into the network only if all of the above decision steps indicate that the new flow will not lead to congestion in the network. We next discuss step 1 of the admission control operation. This involves measurement-based profiling of the neighbor data by the nodes. The remaining steps are described in the subsequent sections.

5.2.1. Profiling the neighbors

Neighbor profiling involves collection of statistics required for the admission control decision. The following statistics about the neighborhood are essential for admission control: RSS and packet reception at each data rate; packets received per second at each data rate; data rate usage on links; and the impact of a node on the Channelbusytime of its neighbors. The following sections discuss the use of these metrics in the design of our admission control scheme. We now discuss the mechanism of collecting data from the neighborhood. Note that the measurement operation is a continuous process that ensures that nodes have up-to-date information about their neighborhood.

In order to minimize the network overhead, we collect all our measurement data from the observed data traffic in the network. The MadWifi driver was modified to track the RSS of the received packets as well as the received and sent packet count at different data rates from all the neighbors. This data is aggregated and periodically communicated to the neighbors. The communication of the aggregated data occurs with a periodic broadcast packet. MDAC needs neighbor statistics from a two-hop neighborhood. The one-hop neighbor statistics obtained during the current cycle of broadcast are, therefore, piggybacked in the next cycle of packet broadcast. Note that it is possible to eliminate the broadcast communication by piggybacking the statistics on data packets.

With the received packet count statistics from neighbors, nodes can determine the packet reception rate at different data rates to each neighbor. Nodes can then construct piecewise linear curves at each data rate using the reception rate and RSS statistics. Nodes also compute the average medium time consumption from the average RSS statistics using the PHY Deferral model proposed by Reis et al. [17]. Ref. [17] provides a detailed description of the PHY Deferral model.

Relying on the observed data traffic has an implication that nodes can only track data rates used by their neighbors. As we will show in the following sections, the data at other rates is not important in our admission control design since these rates are not used.

To summarize, we maintain the following information at each node in the network:

- RSS and reception statistics per neighbor averaged over a time window. We represent the average RSS from a node S at R as \bar{R}_{SR} .
- Piecewise linear curves representing the relationship between RSS and data reception at different data rates. Piecewise curves predict the reception rate at a certain RSS for a given data rate at a node. At a node R , we represent this function of \bar{R}_{SR} and data rate d as $\hat{p}_R(\bar{R}_{SR}, d)$. Note that these curves are periodically updated with the most recent information.
- Packets received per second and data rate statistics of the two-hop neighborhood. We represent total packets per second at a node R as PPS_R ; packets per second from S at R as PPS_{SR} ; and packets per second from S at R at a data rate d as $PPS_{SR}(d)$.
- Fractional impact of a node on Channelbusytime of its neighbors. We represent the fractional impact of a node S on Channelbusytime of a node R as BI_{SR} , and it can be understood as follows: for every y packets transmitted by node S , $BI_{SR} * y$ packets are carrier sensed at node R .
- The data rate that is most likely to be used between a node S and any of its neighbors R , and the expected number of transmissions to send a packet successfully at the data rate. The most likely data rate, DR_{SR} , is the data rate that is used most during the previous interval of time. The expected number of transmissions at a data rate d on link $S - R$ is represented as ETX_{SR}^d .

5.2.2. Channelbusytime estimation

We now describe the design of steps 2 and 3 of MDAC operation. These steps involve estimating the local busy-time and the neighborhood busy-time. Through these estimations, we can ensure the availability of sufficient bandwidth to prevent throttling of packet transmissions due to the admission of a new flow.

5.2.2.1. Local Channelbusytime. As discussed in Section 5.1, Channelbusytime is the fraction of time spent in the TX, RX or ChannelBusy states. Channelbusytime includes the MAC overhead caused by the silence periods due to DIFS, SIFS and backoff. The MAC overhead, t_{MAC} , can be approximated by measuring the number of overheard data and ACK packets. For a network operating on the 2.4 GHz band, we can account for t_{MAC} by promiscuously measuring the packets heard on 802.11b and 802.11g. Let N_b^d, N_g^d, N_b^a and N_g^a represent the number of data packets at 802.11b and 802.11g and the number of ACK packets at 802.11b and 802.11g, respectively, and $DIFS_b, DIFS_g, SIFS_b$ and $SIFS_g$ represents the DIFS values at 802.11b, 802.11g and the SIFS values at 802.11b and 802.11g, respectively. We can then write t_{MAC} as:

$$t_{MAC} = N_b^d * DIFS_b + N_g^d * DIFS_g + N_b^a * SIFS_b + N_g^a * SIFS_g + \text{avgbackoff}(n)$$

The average backoff can be computed as a function of the number of neighbors n at a node [22]. Note that the Channelbusytime estimation can only be a close approximation as it is difficult to model backoff accurately and account for all the packets that cause medium busy-time at a node.

Table 1 shows a comparison of Channelbusytime of two flows A and B with different packet sizes and packets per second at comparable t_{CCA} values. The values are computed for a data rate of 11 Mbps and average backoff of 7. The Channelbusytime difference is about 12%, illustrating that Channelbusytime computed without accounting for MAC overhead can be very optimistic, especially for flows with small packet size.

Let the Channelbusytime requirement of the data and ACK packets of a new flow f to be admitted between nodes S and R be BR_f^b and BR_f^a . The Channelbusytime requirement values can be estimated with the knowledge of the bit rate of the flow, the likely data rate on link $S-R(D_{SR})$ and the expected number of transmissions at data rate D_{SR} . The bit rate information for the flow can be obtained from the application layer. For instance, a typical value of 64 kbps with a packet size of 160 bytes can be assumed for VoIP flows using G.711 codec.

If δ represents the margin for variations, the first check for admission of a flow f from S to R is given by:

$$BR_f^d + BR_f^a + \delta < 1 - \text{Channelbusytime}_S \quad (2)$$

$$BR_f^d + BR_f^a + \delta < 1 - \text{Channelbusytime}_R \quad (3)$$

5.2.2.2. Channelbusytime at neighbors. The Channelbusytime on the neighbor nodes of S and R needs to be assessed to prevent the throttling of packet transmissions at these nodes. Consider a neighbor node N for the sender S . Let $BI_{SN} \geq 0$ and $BI_{NS} \geq 0$ represent the fractional Channelbusytime impacts of one node on the other. Consider that the neighbor node N has $t_f = 0$; its medium time is completely saturated by its transmission and receptions. Let the local admission check at S and R in Eqs. (2) and (3) pass for a new flow f . This indicates that there is sufficient free time to accommodate all the transmissions at S and R . However, this check does not suffice to prevent the transmission at N from being throttled by the new flow f , even for the case of S being able to carrier sense all the packet transmissions from N . This is because additional transmissions from the new flow f contend for the medium with node N 's transmissions. The increase in contention results in increasing the overall transmit duration for the node N . However, since node N is already completely saturated

by the existing transmissions and receptions, node N is likely to experience throttling of its transmissions if the new flow is admitted.

The additional medium contention introduced depends on the fractional Channelbusytime factors BI_{SN} and BI_{NS} . For the best case of $BI_{SN} = 0$, there is no transmission throttling at node N . However, for any $BI_{SN} > 0$, the impact on the transmissions at N depends on the extent of the transmission period that node S 'steals' from node N . This depends on multiple factors: the data rates selected by the nodes, the packet scheduling and the impact of one node on Channelbusytime the other. We consider the free time left at node N if flow f steals all the medium time as predicted by the Channelbusytime impact of node S and R .

If δ represents the margin for variations, the next check for admission of a flow f from S to R is given by:

$$BI_{SN} * BR_f^d + BI_{RN} * BR_f^a + \delta < 1 - \text{Busytime}_N \quad (4)$$

$BI_{SN} * BR_f^d$ represents the Channelbusytime impact of the data packets from S at N , and $BI_{RN} * BR_f^a$ represents the Channelbusytime impact of the ACK packets from R at N .

5.2.3. Collisions

The final piece of the framework involves estimation of the impact of packet collisions in the neighborhood: specifically, on the existing traffic and the new flow to be admitted, representing the steps 4 and 5 in the admission control operation described in Section 5.2.

5.2.3.1. Impact on existing traffic. We classify packet collisions into two categories. The first type of collision occurs when two carrier sensing neighbors collide due to random medium access behavior of 802.11. In the second category, the packet transmission of two nodes outside the carrier sense range of each other collide due to simultaneous medium access. This is the classic hidden terminal problem. Steps 2 and 3 of MDAC operation ensure that the available medium time between all contending nodes is sufficient to accommodate the new flow. This Channelbusytime condition keeps the collision impact due to random medium access under control. However, we must account for the collisions due to hidden terminal medium access.

We now consider the collision impact of a new flow f at node S on a neighbor N . Assume that all the nodes from which N receives traffic lie outside the carrier sense range of node S . In other words, node S is a hidden terminal for all packet receptions at N . Let C represent the cycle time for the new flow f at S . The cycle time represents the average inter-packet time duration. The cycle time must account for the expected number of transmissions ETX_{SR}^d on the link $S-R$ at the likely data rate DR_{SR} . For instance, a VoIP flow generating packets every 20 ms on a link with $ETX = 2$ has a cycle time of 10 ms. Let $T_N(d)$ represent the average time duration of a packet on the medium at node N operating at data rate d . Let DS represent the data rate

Table 1
Channelbusytime comparison of two flows

Flow	Pkts per second	Pkt size	t_{CCA}	Busytime
A	1000	160	61.24%	81.24%
B	390	1500	61.89%	69.69%

set consisting of all the data rates observed at the received packets at node N . If p_N^d represents the fraction of the packets received at data rate d at node N , we can approximate the probability of overlap of packets from flow f with the packets at N as follows:

$$P_{SN} = \frac{\sum_{i \in DS} p_N^i * PPS_N * C * (T_N(i) + T_S(DR_{SN}))}{C}$$

P_{SN} represents the probability of packet overlap of the new flow on all receptions at node N . The portion $p_N^i * PPS_N * C$ represents the number of packets seen at data rate i in the time duration C . The factor $T_N(i) + T_S(DR_{SN})$ indicates that a packet from S that occupies $T_S(DR_{SN})$ medium time can overlap with a packet at N occupying $T_N(i)$ medium time if it starts anywhere within a time duration $T_S(DR_{SN})$ before the start of the packet at N upto the duration of the packet. For a sender node M , if DR_{MN} represents the most likely data rate on the link $M - N$, the probability of packet overlap of the flow f on a link $M - N$ can then be computed as:

$$P_{SN}^{MN} = \frac{PPS_{MN} * T_M(DR_{MN}) * P_{SN}}{\sum_{i \in DS} p_N^i * PPS_N * T_N(i)}$$

where PPS_{MN} represents the packets received per second from node M at N and PPS_N is the total packets received per second at node N . P_{SN}^{MN} represents the fraction of packets that will overlap with the new flow f .

Note that the above analysis assumes that the node S is out of the carrier sense range of all the senders. We can account for the partial carrier sense behavior between node S and the senders to node N by considering the fractional decrease in the packets per second that are susceptible to collisions. This can be done by replacing the traffic fraction at data rate i , $p_N^i * PPS_N$, in the equation for P_{SN} with $p_N^i * \sum_{K \in senders} (PPS_{KN} * (1 - BI_{KS}))$. With this modification, we account for the overlap probability for the fraction of packets that are not carrier sensed at S and hence are susceptible to hidden terminal collision.

With the Channelbusytime impact accounted for, the probability of overlap per link, P_{SN}^{MN} can then be written as:

$$P_{SN}^{MN} = \frac{(1 - BI_{MS}) * PPS_{MN} * T_M(DR_{MN}) * P_{SN}}{\sum_{i \in DS} p_N^i * (\sum_{K \in senders} PPS_{KN} * (1 - BI_{KS})) * T_N(i)}$$

The packet reception probability, PR_{MN}^t , for link $M - N$ can now be written as follows:

$$PR_{MN}^t = P_{SN}^{MN} * \hat{p}_R(\bar{R}_{MN}^t, d) + (1 - P_{SN}^{MN}) * \hat{p}_R(\bar{R}_{MN}, d)$$

where \hat{p}_R is the piecewise function at node R and \bar{R}_{MN}^t represents the effective RSS taking into account the overheard RSS of the interfering packets, as given by the Reis measurement model [17]:

$$\bar{R}_{MN}^t = \bar{R}_{MN} - \delta_N(\bar{R}_{SN})$$

δ_N is the SINR threshold for packet reception at node N at the most likely data rate DR_{MN} .

We now have the condition to check whether the impact of packet collisions by the new flow on neighbor packet receptions is below an allowed threshold:

$$PR_{MN} - PR_{MN}^t < \delta(d, ETX_{MN}^d) \quad (5)$$

Note that δ is a function of the current data rate d and the expected number of transmissions at d on the link $M - N$. We have consciously omitted the impact of ACK packets in the above analysis for ease of understanding. It is straightforward to incorporate ACK packet collision by similar reasoning. Note that ACK packets cause hidden terminal collisions at all neighbors of the receiver R that lie outside carrier sense range of the sender S . This is because neighbors of the node S that can overhear the data packet set the NAV value so as to not collide with the ACK packet. The equations do not account for the impact of packet size. It is straightforward to incorporate the average packet size measured at the nodes.

5.2.3.2. Impact on the new flow. The above analysis quantifies the impact of packet collisions on the existing traffic in the medium. The analysis for the impact of packet collisions on the new flow is similar.

To summarize, a new flow is admitted into the network only if all the operational steps represented by Eqs. (2)–(5) indicate that the new flow will not lead to congestion in the network. We next discuss the implementation and evaluation of MDAC over the UCSB MeshNet testbed.

6. Implementation

We implemented our MDAC system on the UCSB MeshNet. Our implementation consists of two main modules at each node in the network: a measurement-based profiling module and the decision control framework. The profiling module collects and maintains the statistics at each node about its neighbors. The decision control framework makes flow admission decisions based on these statistics.

6.1. Measurement-based profiling

Our measurement framework consists of modifications to the Linux MadWifi driver code along with a user-space module to collect and exchange statistics between nodes. We first explain how the neighbor packet reception and RSS statistics are tracked at a node. Our admission control scheme requires reception statistics at available data rates. As discussed in Section 4.3, maintaining these statistics through probe messages at each rate can lead to significant overhead in the network. We therefore leverage the application packet statistics maintained by the rate selection algorithm. The implementation of the SampleRate algorithm, obtained as a module of the MadWifi driver, tracks per-neighbor statistics, which include the number of packets sent and the average number of packet transmission

attempts at different data rates. This information is periodically extracted and broadcast every 10 s.

We modified the MadWifi driver to track the average RSS of all received packets and the number of received packets from each neighbor at observed data rates. With this local information and the packet statistics information obtained from the broadcast packets, nodes can compute the packet loss and average RSS from their neighbors at different data rates. As discussed in Section 5, nodes track packets-per-second statistics within a two-hop neighborhood. This is done by piggybacking packet statistics obtained from one-hop neighbors during the current cycle in the next cycle of packet broadcasts.

To construct the RSS–Packet Reception piecewise curves at each node, we leverage the 802.11 specification datasheet. The datasheet specifies the receiver sensitivity of the radio at different data rates. The nodes are bootstrapped with this sensitivity value as the minimum RSS required for successful packet reception at a given data rate. Actual packet reception and the RSS statistics obtained through measurements are then taken into account to incrementally build the piecewise linear curves. The time to compute the RSS–Packet Reception piecewise curve at a node can vary based on the amount of application traffic at the node. This is because instead of generating additional traffic, we leverage the application traffic and the observed data rates in the medium for mapping packet reception as a function of the signal strength of the packets. At any instance, however, the piecewise curves reflect the most recent measurements.

The ETX and data rate information for a node pair is obtained from SampleRate statistics using the `proc` file-system interface. The last piece of building the neighbor profile involves measuring the impact of a node on Channelbusytime of its neighbor. The Channelbusytime impact is computed from the average RSS statistics of the received packets as described in Section 5.2.1.

6.2. Decision control framework

The decision control framework is implemented as a user space library that can be imported into the application code. As discussed in Section 5, we implement the four operations of estimating local medium time, neighbor Channelbusytime, packet collisions caused by the new flow and packet collision rate of the flow.

For our evaluations, we use a Busy-time margin of 10% for the δ parameter in Eqs. (2)–(4). In other words, medium Busy-time is considered fully utilized at 90% usage. The above value of δ permits aggressive admission of flows and improves the medium utilization. Computation of δ for Eq. (5) is non-trivial because the reaction of a node to packet loss on a link depends on the rate selection algorithm. In our implementation, we set parameters for use with the SampleRate algorithm. SampleRate switches to a data rate that is expected to provide the lowest transmission time for a successful packet transmission. We can thus

compute the packet reception rate required at each data rate before a switch to a lower data rate occurs. For instance, at 11 Mbps, packets can suffer up to about 50% loss before a switch to 5.5 Mbps occurs. We restrict flows when the predicted collision losses at the current data rate reach within a margin of $\delta = 10\%$ of the packet losses at which SampleRate switches to a lower data rate.

7. Evaluation

We evaluate our admission control scheme on the UCSB MeshNet testbed, as described in Section 2.1. The focus of our evaluation is to understand the Channelbusytime impact and collision prediction, the ability of MDAC to limit flows in the network, and to ensure QoS to VoIP flows.

7.1. Channelbusytime prediction

We first understand the accuracy of Channelbusytime prediction in the presence of traffic on the testbed. This helps us understand the effectiveness of computing local and neighborhood Channelbusytime as part of the admission control metric.

In these experiments our objective is to understand the accuracy of Channelbusytime prediction. We do not yet perform admission control, and hence we expect packet collisions, particularly from hidden terminals, to affect the Channelbusytime prediction. Packet losses due to collisions lead to variation in the data rate used by the rate selection algorithm. This can in turn affect the Channelbusytime usage at the nodes that overhear the traffic. The prediction is thus expected to be less accurate in the scenarios when the Channelbusytime is affected by packet collisions.

We adopt the following procedure to filter the results in situations where we expect the variation in data rate due to collisions to have caused a significant impact on the Channelbusytime at nodes. With the collected statistics, we can isolate instances where nodes outside the carrier sense range of the node that has initiated a flow experience significant variation in the Channelbusytime for the duration of the flow. In other words, we separate instances where nodes hidden from the initiator node experience significant Channelbusytime variation during the flow. One of the factors that may have caused this variation is the packet collisions that result from the admission of the new flow. Other factors include link quality variations or traffic variations. However, the traffic on our testbed does not vary significantly due to the constant bit rate used in the experiments. We therefore believe that packet collisions account for a majority of these instances of variation in Channelbusytime.

7.1.1. Methodology

At a central server, we generate a flow sequence file that lists a random sequence of 50 one-hop flows between node pairs in the network and indicates the initiation time and

duration of the flow. The average inter-flow arrival time is 20 s and the average duration of each flow is 300 s. The random sequence of flows enables us to study the busy-time behavior with different levels of network utilization. The flow sequence file is distributed to all the nodes in the network. Before the start of the experiment, the nodes in the testbed are time-synchronized with a common NTP server to the accuracy of about 5 ms. The nodes initiate 64 kbps UDP flows using a fixed packet size of 160 bytes in accordance with the initiation and duration time specified in the sequence file. Additionally, all nodes collect the available neighborhood Channelbusytime statistics 2 s prior to and 10 s after a flow initiation between any pair of nodes. The nodes also record the predicted value of increase in Channelbusytime for each flow prior to its initiation.

We repeat the experiment five times and collect the Channelbusytime statistics from all nodes. With the help of the Channelbusytime statistics collected prior to every flow initiation, we compute the deviation in the observed Channelbusytime from the predicted increase in Channelbusytime of a flow.

7.1.2. Results

Fig. 7 plots a CDF of the deviation of the predicted medium Channelbusytime from the observed Channelbusytime for all the results and the filtered results. Perfect prediction of the Channelbusytime would result in a vertical line at 0 Channelbusytime Deviation on the x -axis. The median Channelbusytime variation for the filtered results is about 6%, while it is about 18% for the complete set of results. We make two inferences from these results. First, collisions have a significant effect on the accuracy of Channelbusytime prediction. This implies that the performance of admission control schemes that consider only medium busy-time to limit flows may suffer due to packet collisions in the network. Second, the Channelbusytime prediction, in the absence of hidden terminal collisions, is fairly accurate (<10% deviation).

Fig. 8 plots the deviation in Channelbusytime prediction at different Channelbusytime values. The 45° line represents the zero-deviation or perfect prediction scenario. The error-bars indicate the deviation about the mean pre-

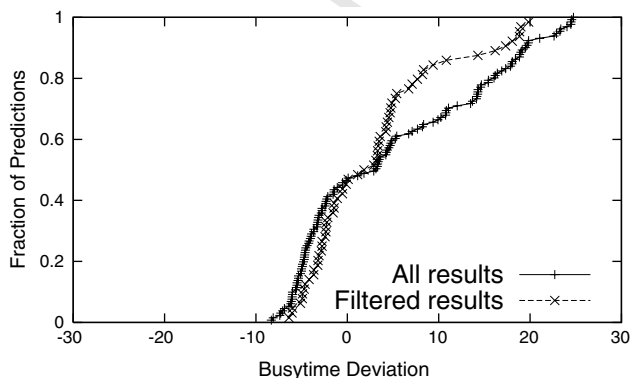


Fig. 7. CDF of predicted Channelbusytime error.

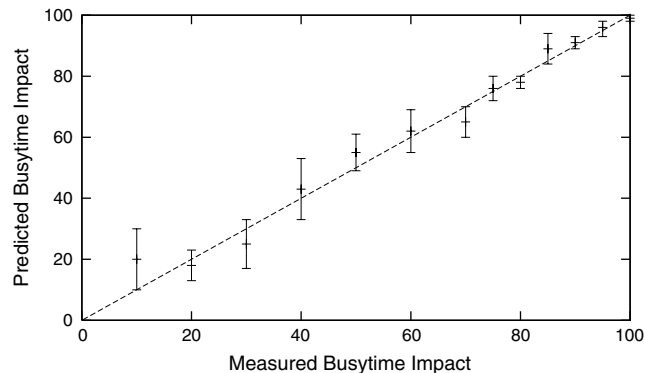


Fig. 8. Deviation of the predicted Channelbusytime impact.

dicted value of Channelbusytime. We observe that the prediction is more accurate when the Channelbusytime impact is higher. This is encouraging because the nodes that experience higher Channelbusytime impact are more important for admission control purposes. Nodes with less than 30–40% Channelbusytime impact are less important and the higher error rates in prediction in this region means that, at times, the admission control scheme can be pessimistic.

7.2. Admission control

We now evaluate the effectiveness of MDAC in performing admission control. In particular, we study the throughput performance of the flows in a network that uses MDAC.

7.2.1. Methodology

The basic methodology of network setup and the flow initiation is similar to the methodology used in Section 7.1. We initiate 30 flows with a flow interval rate of 10 s between random sets of one-hop node pairs on the testbed. Each flow is a 64 kbps constant bit rate UDP flow with fixed packet size of 160 bytes and lasts for 300 s. We track the average throughput (in packets per second) and the average delay experienced by the individual flows. We study the performance of MDAC in two system configurations: in the first, we consider only the Channelbusytime estimation for admission control decisions; in the second, we consider both the Channelbusytime estimation and the collision estimation for admission control decisions. The second scenario represents the complete MDAC solution. Note that we have shown the performance results for these experiments in the absence of an admission control system in Section 2.

7.2.2. Results

Fig. 9a shows a timeline of the throughput performance of 15 admitted flows using only the Channelbusytime estimate, i.e., without collision estimation. We find that the throughput performance of four flows is significantly affected (>25% drop in throughput). Repeated experiments with higher Channelbusytime margins (δ) of up to 50% did

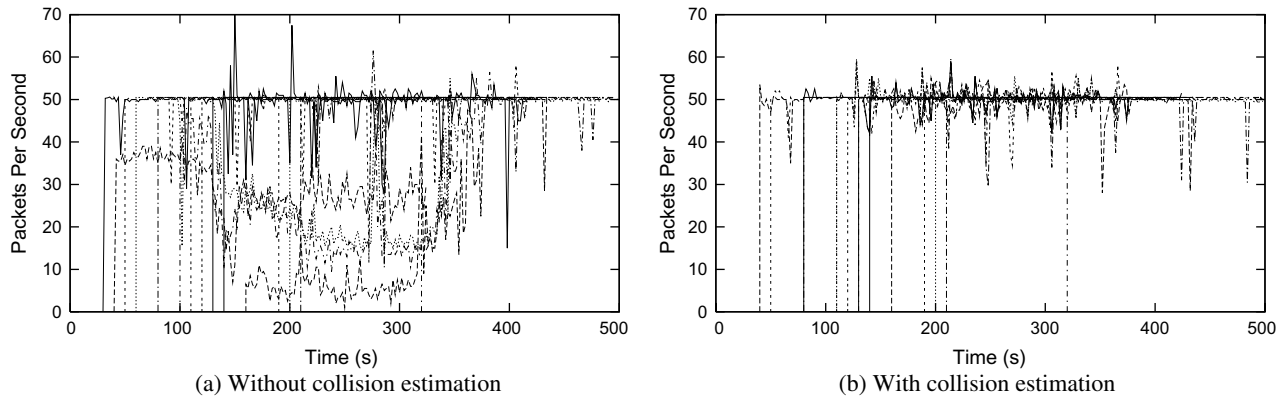


Fig. 9. Per-flow throughput with MDAC.

not prevent the throughput reduction of these flows. The average packet delay for the four affected flows is 1.2, 2.3, 1.5 and 2.1 s. These results illustrate that Channelbusy-time predictions alone do not suffice for admission control of flows. Fig. 9b shows the timeline of performance using the complete MDAC system, i.e., with both the Channelbusytime and collision estimates. The graph shows that 12 flows were admitted and the network could sustain all the admitted flows. Measurements showed that the average packet delay of all the 12 admitted flows was under 60 ms. This result shows that MDAC is effective in limiting the number of flows in the network and ensuring bandwidth availability to admitted flows.

7.3. Impact of flow arrival rate

We now explore the relationship between the average inter-arrival time of the flows and the frequency of broadcast packets that are used for communication of admission control metrics among neighbors. MDAC relies on the accuracy and consistency of the neighbor profile to arrive at correct flow admission decisions. Therefore, if the flow arrival rate is greater than the rate of broadcast packets, we expect the performance of MDAC to be affected. We conduct the following experiment to investigate this relationship.

7.3.1. Methodology

The methodology of experiment is the same as that discussed in Section 7.2 with the exception that the flow arrival rate is 5 s. Broadcast packets continue to be transmitted at 10-s intervals.

7.3.2. Results

Fig. 10 shows the throughput performance of the flows. The failure to limit flows can be attributed to the delay in communication of updated information to the neighbors as compared to the flow arrival interval. For instance, if two flows that cause significant packet collisions at each other due to the senders being hidden terminals arrive within a span of 5 s, both the sender nodes make admission

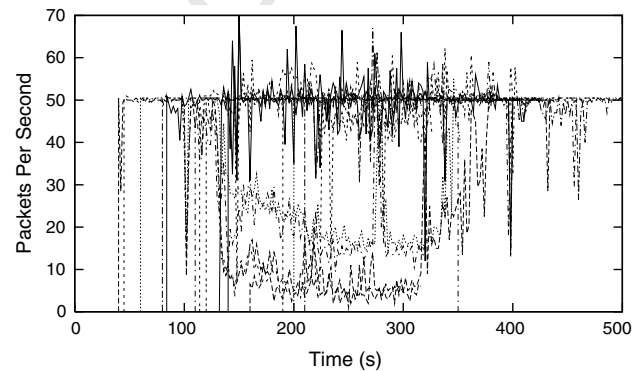


Fig. 10. Per-flow throughput with 5 s inter-flow arrival time.

control decisions based on the old information available from the network. Thus, the framework fails to sustain the throughput performance of all the admitted flows on the testbed.

7.3.3. Discussion

There are multiple reasons for the occurrence of false admissions in the above illustration. It is possible to minimize such occurrences with appropriate choice of parameters. For the purpose of evaluation of MDAC, we chose a broadcast interval rate of 10 s to communicate the aggregate rate and packet statistics. This rate of information exchange accounts for less than 1% of the medium time for a network where the nodes have an average of 10 neighbors and, thus, represents a sparse exchange of information. The broadcast interval rate should be chosen based on a tradeoff between the acceptable levels of control traffic and the expected flow arrival rate. A higher rate of information exchange will significantly reduce instances of false admissions.

Another factor is the choice of values for the Channelbusytime and collision margins allowed during admission control. In our evaluations, the Channelbusytime margin is 10% and the packet collision margin to prevent data rate switch by the SampleRate is 10%. Reserving additional bandwidth in the network with higher margins makes the

admission control system more conservative, thereby reducing false admission of flows in the network. These parameters can be tuned for different traffic and network scenarios to minimize the impact of false admissions.

7.4. Ensuring QoS

We now evaluate the performance of MDAC in terms of providing the desired quality of service for the example application of VoIP. A practical guideline to evaluate the quality of VoIP flows in a network includes the following two constraints: a network delay budget of 80 ms, and packet loss rate below 5% assuming the presence of error concealment algorithms [19].

7.4.1. Methodology

We follow a similar methodology as in Section 7.2 with a flow interval rate of 10 s and duration of 300 s. Additionally, we implement a handshaking protocol that mimics the call-setup phase before the start of each VoIP flow. The duration of the experiment is 30 minutes and we run seven trials of the experiment with different flow sequence files. For each flow topology, the experiment is performed with the MDAC system as well as without admission control.

7.4.2. Results

Fig. 11 shows a comparison of the number of flows admitted with admission control (AC) and no admission control (No AC) for different flow topologies. Each topology corresponds to a different experiment trial. The figure also shows the number of flows for which the QoS requirements were satisfied with admission control and no admission control. Of the possible 180 flows, the maximum number of flows that were admitted during any experiment is only 93. This is because heavy congestion causes failure of the call-setup handshake and prevents establishment of the flow. The results show that without admission control in place, only a small fraction of the flows have their QoS requirements satisfied. On the other hand, MDAC allows initiation of only about 50–60% of the number of flows initiated without admission control. More impor-

tantly, the QoS requirements are met for about 95% of the admitted flows. The graph also shows the maximum number of flows that could be supported with QoS. This data was obtained by iteratively repeating the admission of flows and obtaining the maximum number of flows that were supportable while the QoS requirements of all flows were met. We observe that MDAC can be slightly pessimistic in that it supports about 90% of the system maximum.

8. Discussion

In this section, we discuss some of the future challenges and the general applicability of MDAC.

8.1. Mixed traffic and traffic priority

We have not considered handling mixed traffic and traffic priorities in our scheme. While the use of 802.11e-type priorities helps improve the medium access behavior of voice flows in a mixed network, data traffic can still impose a significant performance penalty on voice flows when the resources are constrained. Intelligent throttling of data flows and estimation of bandwidth availability for voice flows in the presence of data traffic in a realistic environment needs more research. We believe that our work is a critical step towards the achievement of complete control of user traffic in a real network.

8.2. Multi-hop networks

The design and evaluations presented in this paper have only considered single-hop flows. MDAC can be extended to multiple hops by executing the operational steps at all the nodes along a new flow path. The basic framework of Channelbusytime computation and collision estimation remains the same. However, the nodes along the path need to coordinate to compute the contention count [23] imposed at each neighbor in the network.

8.3. Applicability

Our scheme has been shown to work well in indoor static wireless networks. We believe that our scheme will also perform well in any general indoor setting. However, without real experiments, it is difficult to predict the applicability in a outdoor wireless network scenario. Outdoor wireless medium characteristics exhibit significant differences from indoor networks. For instance, due to the large delay spread, the RSS–Packet Reception relationship might not hold as well in these networks. As future work, we intend to investigate the operation of MDAC in outdoor settings.

9. Related Work

The problem of ensuring QoS in 802.11-based wireless networks presents several challenges and has been a topic

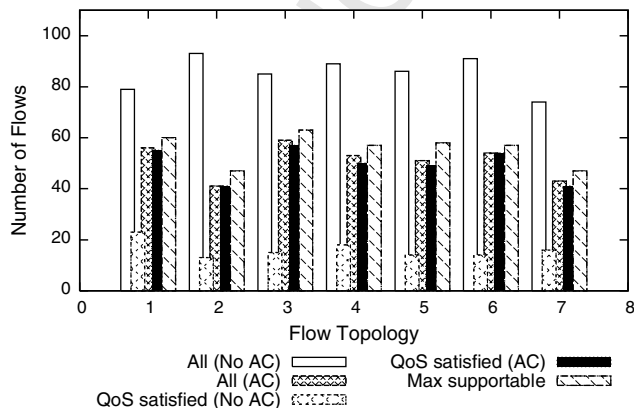


Fig. 11. Number of VoIP flows admitted.

of active research for several years. The shared nature of the medium and the contention-based access of the 802.11 protocol makes the problem of ensuring the QoS requirements of delay-sensitive and bandwidth-sensitive multimedia applications a challenging problem.

In prior research, TDMA-style approaches [15,14,24,5] have been proposed to enable fine-grained control of the medium and provide QoS guarantees to real-time applications. The controlled access of the medium simplifies the problem of reserving bandwidth on a per-flow basis. These approaches, however, require effective synchronization among the nodes in the network and are thus difficult to realize in a real-world network. Other QoS solutions [13,2] consider the admission control system as an important component of a generic QoS framework, but do not investigate the problem of bandwidth reservation in a shared medium in sufficient detail.

Some QoS solutions, such as [20,4], propose a joint admission control and routing scheme for ad hoc wireless networks. They provide detailed computations that can be used to estimate available bandwidth in a multi-hop wireless network. The bandwidth computations for a node assume a 100% Busytime impact for all the neighboring nodes. However, in our work we show that this assumption leads to overestimation of the bandwidth requirements in the network.

Contention-aware admission control protocol [23] and perceptive admission control [11] attempt to address the challenges of admission control in a multi-hop wireless network. These approaches focus their attention on the problem of communication with neighbors in the carrier sense range and propose solutions that cannot be implemented with current hardware. In contrast, we have demonstrated the possibility of communication with carrier sense neighbors using commodity hardware.

SoftMAC [19] proposes a software framework at that employs coarse-grained control to regulate network load and ensure QoS. The authors suggest the use of a Fraction of Air Time metric to estimate the bandwidth available at a link. SoftMAC, however, does not account for either the impact of collisions or the partial Busytime impact discussed in this paper.

To the best of our knowledge, our work is the first admission control solution that relies on actual network measurements for its decision-making framework. By using the measured Busytime and link quality metrics during the admission control decision, MDAC accounts for the impact of collisions and the partial busy-time impact. This enables MDAC to successfully limit the number of flows and prevent congestion in a real network.

10. Conclusion

We present MDAC, a measurement-driven admission control framework that leverages wireless characteristics for flow control in a static wireless network. We performed an extensive evaluation of our solution on the 25 node

UCSB MeshNet to understand some of these wireless characteristics and to show that the proposed scheme works well to provide QoS requirements to real time applications such as VoIP.

With the growing popularity and increasing usage of 802.11 wireless networks, there is an urgent need for mechanisms to prevent network breakdown in the presence of large traffic volumes. While our work is a critical first step towards realistic flow control in wireless networks, we believe that much work remains to address the challenges of a mixed traffic network and make admission control robust enough to handle all network and traffic scenarios.

References

- [1] D. Aguayo, J. Bicket, S. Biswas, G. Judd, R. Morris, Link-level measurements from an 802.11b mesh network, in: Proceedings of SIGCOMM (Portland, OR, August 2004).
- [2] G.-S. Ahn, A. Campbell, A. Veres, L.-H. Sun, Supporting service differentiation for real-time and best-effort traffic in stateless wireless ad hoc networks (SWAN), *IEEE Transactions on Mobile Computing* 1 (3) (2002) 192–207.
- [3] J. Bicket, Bit-rate selection in wireless networks, Master's thesis, Massachusetts Institute of Technology, 2005.
- [4] L. Chen, W.B. Heinzelman, QoS-aware routing based on bandwidth estimation for mobile ad hoc networks, *IEEE JSAC* 23 (3) (2005) 561–572.
- [5] T.-W. Chen, J. Tsai, M. Gerla, QoS routing performance in multihop, multimedia, wireless networks, in: Proceedings of IEEE International Conference on Universal Personal Communications (San Diego, CA, June 1997).
- [6] D.S.J. De Couto, D. Aguayo, J. Bicket, R. Morris, A high-throughput path metric for multi-hop wireless routing, in: Proceedings of MobiCom (San Diego, CA, October 2003).
- [7] R. Draves, J. Padhye, B. Zill, Comparison of routing metrics for static multi-hop wireless networks, in: ACM International Conference on Special Interest Group on Data Communication (Portland, OR, March 2004).
- [8] R. Draves, J. Padhye, B. Zill, Routing in multi-radio, multi-hop wireless mesh networks, in: Proceedings of MobiCom (Philadelphia, PA, September 2004).
- [9] P. Gupta, P.R. Kumar, The capacity of wireless networks, *IEEE Transactions on Information Theory* IT-46 2 (2000) 388–404.
- [10] T. Henderson, D. Kotz, I. Abyzov, The changing usage of a mature campus-wide wireless network, in: Proceedings of MobiCom (Philadelphia, PA, September 2004).
- [11] I.D. Chakeres, E.M. Belding-Royer, PAC: perceptive admission control for mobile wireless networks, in: Proceedings of QShine (Dallas, TX, October 2004).
- [12] A.P. Jardosh, K. Mittal, K.N. Ramachandran, E.M. Belding, K.C. Almeroth, IQU: practical queue-based user association management for WLANs, in: Proceedings of MobiCom (Los Angeles, CA, September 2006).
- [13] S.-B. Lee, G.-S. Ahn, X. Zhang, A. Campbell, INSIGNIA: an IP-based quality of service framework for mobile ad hoc networks, *Journal of Parallel and Distributed Computing* 60 (4) (2000) 68–76.
- [14] C.R. Lin, J.-S. Liu, QoS routing in ad hoc wireless networks, *IEEE JSAC* 17 (8) (1999) 1426–1438.
- [15] C. Lin, Admission control in time-slotted multihop mobile networks, *IEEE JSAC* 19 (10) (2001) 1974–1983.
- [16] J. Padhye, S. Agarwal, V.N. Padmanabhan, L. Qiu, A. Rao, B. Zill, Estimation of link interference in static multi-hop wireless networks, in: Proceedings of Internet Measurement Conference (Berkeley, CA, October 2005).

- 1556 [17] C. Reis, R. Mahajan, M. Rodrig, D. Wetherall, J. Zahorjan, Measure- 1571
1557 ment-based models of delivery and interference in static wireless 1572
1558 networks, in: Proceedings of SIGCOMM (Pisa, Italy, September 2006). 1573
1559 [18] S.H.Y. Wong, S. Lu, H. Yang, V. Bharghavan, Robust rate 1574
1560 adaptation for 802.11 wireless networks, in: Proceedings of MobiCom 1575
1561 (Los Angeles, CA, September 2006). 1576
1562 [19] H. Wu, X. Wang, Y. Liu, Q. Zhang, Z. Zhang, SoftMAC: layer 2.5 1577
1563 MAC for VoIP support in multi-hop wireless networks, in: Proceed- 1578
1564 ings of SECON (Santa Clara, CA, September 2005). 1579
1565 [20] Q. Xue, A. Ganz, Ad hoc QoS on-demand routing (AQOR) in mobile 1580
1566 ad hoc networks, *Journal of Parallel and Distributed Computing* 63 1581
1567 (2) (2003) 154–165. 1582
1568 [21] K. Xu, M. Gerla, S. Bae, How effective is the IEEE 802.11 RTS/CTS 1583
1569 handshake in ad hoc networks, in: Proceedings IEEE GLOBE- 1584
1570 COM'02 (Taipei, Taiwan, November 2002). 1585
1586 [22] X. Yang, N. Vaidya, On the physical carrier sense in wireless ad hoc 1571
1572 networks, in: Proceedings of INFOCOM (Miami, FL, March 1573
2005). 1574
1575 [23] Y. Yang, R. Kravets, Contention-aware admission control for ad hoc 1576
1577 networks, *IEEE Transactions on Mobile Computing* 4 (4) (2005) 363– 1578
377. 1579
1580 [24] C. Zhu, M. Corson, QoS routing for mobile ad hoc networks, in: 1581
1582 Proceedings of ICC (Miami, FL, June 2002). 1583
1584 [25] Cisco 7900 Series IP Phones, December 2006. 1585
1586 [26] Global Wi-Fi usage grew 111% in past 10 months, December 2006. 1586
1587 [27] Intersting [MIT RoofNet], December 2006. 1587
1588 [28] MadWifi, December 2006. 1588
1589 [29] OpenHAL – MadWifi, December 2006. 1589
1590 [30] UCSB MeshNet, December 2006. 1590
1591 [31] WiFi Outlook Cloudy in Mountain View, December 2006. 1591

UNCORRECTED PROOF