Available online at www.sciencedirect.com
ScienceDirect
Ad Hoc Networks 5 (2007) 228–253
Ad Hoc Networks
www.elsevier.com/locate/adhoc

Throughput guarantees for multi-priority traffic in ad hoc networks

Yaling Yang *, Robin Kravets

Department of Computer Science, University of Illinois at Urbana-Champaign, 201 North Goodwin Avenue, Room 3107, Siebel Center, Urbana, IL 61801-2302, United States

Received 4 May 2005; received in revised form 6 October 2005; accepted 2 November 2005
Available online 6 December 2005

Abstract

In this paper, we present MPARC (Multi-Priority Admission and Rate Control), a novel joint admission control and rate policing protocol for multi-priority ad hoc networks. MPARC is based on our novel bandwidth allocation model, which captures the bandwidth allocation for saturated, unsaturated and semi-saturated networks. MPARC guarantees that the throughput of admitted realtime flows will not decrease due to later arriving realtime flows with equal or lower priorities or due to best effort flows. MPARC achieves this goal by performing accurate admission control on every newly arriving realtime flow and appropriate rate policing on all best effort traffic. Through simulation, we demonstrate that MPARC has better performance than existing approaches.

Keywords: Admission control; Bandwidth allocation; IEEE 802.11; Ad hoc networks; QoS

1. Introduction

The fast spread of small wireless computers has enabled the design and deployment of wireless ad hoc networks. Typical applications proposed for such networks include both realtime and non-realtime applications. While realtime applications, such as conversational audio/video conferencing or on-demand multi-media retrieval, require quality of service (QoS) guarantees for effective communication, best effort applications, such as file transfer, are more tolerant to changes in bandwidth and delay and generally always have backlogged packets for transmission. Supporting both types of applications in an ad hoc network is challenging due to the shared nature of the underlying wireless communication channel. The goal of our research, hence, is to provide QoS guarantees, especially throughput guarantees, for realtime traffic in the presence of best effort traffic and at the same time achieve efficient network utilization.

Providing QoS support in ad hoc networks requires support from the MAC layer to regulate access to the wireless channel. Given this tight coupling, most QoS schemes are designed for a specific MAC layer scheme. In this paper, we focus on

* Corresponding author. Tel.: +1 217 244 7116.
E-mail addresses: yyang8@cs.uiuc.edu (Y. Yang), rhk@cs.uiuc.edu (R. Kravets).
URLs: http://www.ews.uiuc.edu/~yyang8 (Y. Yang), http://www-sal.cs.uiuc.edu/~rhk/ (R. Kravets).

1570-8705/$ - see front matter
doi:10.1016/j.adhoc.2005.11.002

networks based on IEEE 802.11 [21] types of MAC protocols. While IEEE 802.11 is often proposed for ad hoc networks due to its wide availability and simple and robust contention-based access mechanism, IEEE 802.11 does not provide any assurance for the throughput of flows. Recently, it has been proposed to extend IEEE 802.11 to support service differentiation. Such extensions include IEEE 802.11e [14] and the work by Aad and Castelluccia [1]. These extensions divide traffic into different classes and use different contention related parameters (e.g., minimum contention window size, maximum MAC frame size, etc.) to provide service differentiation between the classes. However, these extensions still do not provide any guarantees for the throughput of realtime flows. As the wireless channel becomes overloaded and the number of competing flows increases, the bandwidth share of each flow may decrease. Different from these extensions, our focus is to support throughput guarantees in ad hoc networks. We assume that either IEEE 802.11 or its extensions are used at the MAC layer.

QoS support for realtime flows in ad hoc networks requires two main components: admission control and rate policing. Admission control must prevent new realtime flows from consuming too many resources and disrupting the guarantees made to existing realtime flows. On the other hand, considering that ad hoc networks are proposed for search and rescue environments, admission control must also classify and prioritize realtime traffic so that a new important flow will not be blocked due to existing lower priority flows. For rate policing, the sending rate of best effort traffic must be controlled to prevent it from degrading the QoS of existing realtime flows. Essentially, best effort traffic is given a lower priority than realtime traffic.

Based on the above requirements, the goal of our research is to provide an effective multi-priority based admission control protocol for realtime traffic and a rate policing protocol for best effort traffic for wireless ad hoc networks based on IEEE 802.11 and its extensions to service differentiation (e.g., IEEE 802.11e [14] and work by Aad and Castelluccia [1]). Our joint admission control and rate policing protocol, MPARC (Multi-Priority Admission and Rate Control), guarantees that the throughput of an admitted realtime flow can be maintained and will not be disrupted by newly arriving realtime flows with equal or lower priorities or by best effort flows. A higher priority flow, however, may be admitted by MPARC, even if this higher priority flow reduces the throughput of some existing lower priority realtime flows and best effort flows. The rate policing component of MPARC also ensures that best effort traffic does not hurt any existing realtime flows and is only allowed to fill the bandwidth that is not used by realtime traffic.

The design of MPARC has the following unique contributions. First, we reveal that to ensure throughput guarantees to a flow, the available bandwidth of the flow must be accurately estimated. Due to the shared nature of wireless medium, the available bandwidth must be decomposed into two components: local achievable bandwidth and neighborhood available bandwidth. Second, we propose a novel bandwidth allocation model that captures bandwidth sharing between competing flows in all possible network states so that both the local achievable bandwidth and the neighborhood available bandwidth of a flow can be accurately estimated. Third, we show that based on the accurate knowledge of local achievable and neighborhood available bandwidth, throughput guarantees for realtime traffic can be maintained by making priority-based admission control decisions about realtime traffic and controlling the rate of best effort traffic.

The rest of the paper is organized as follows. In Section 2, we present the importance of estimating available bandwidth and introduce its two components: local achievable bandwidth and neighborhood available bandwidth. In Section 3, we briefly review IEEE 802.11 and its extensions for service differentiation. In Section 4, we introduce our novel bandwidth allocation model for a network where all nodes are in each other's carrier sensing range, which we refer to as *a single-hop network* as opposed to multi-hop networks, where nodes may be outside of each other's carrier-sensing range. Section 5 shows how this model can be extended to multi-hop networks as a heuristic to predict available bandwidth, including estimation of both local achievable bandwidth and neighborhood available bandwidth. Section 7 discusses the sources of inaccuracy caused by the extension. Section 6 discusses how admission control and rate policing can be performed based on the knowledge of available bandwidth. Section 7 addresses how multi-hop networks may affect the accuracy of available bandwidth prediction. In Section 8, we analyze the limitations of using other bandwidth estimation models. Section 9 evaluates the performance of MPARC and compares it with existing approaches. Section 10 concludes our work and discusses future directions.

2. Available bandwidth estimation

Admission control for realtime traffic and rate policing for best effort traffic are essentially a problem of determining available bandwidth. For admission control of a new realtime flow, the available bandwidth of the flow is the maximum transmission rate that the new realtime flow is able to achieve subject to the constraint that the new flow does not degrade the throughput of existing equal or higher priority flows. If there is not enough available bandwidth for a new realtime flow, the flow is rejected by admission control. For rate policing of best effort traffic at a node, the available bandwidth is defined as the maximum transmission rate of best effort traffic that does not degrade the throughput of any existing realtime flows. Rate policing controls the aggregated sending rate of all best effort flows below the available bandwidth to ensure the throughput of realtime flows.

Due to the shared nature of the wireless medium and the lack of central control in ad hoc networks, both the throughput of a new flow and the throughput of existing flows are determined by their contention at the MAC layer. Therefore, the available bandwidth must be decomposed into two components, *local achievable bandwidth* and *neighborhood available bandwidth*, where the local achievable bandwidth captures the throughput of the new flow and the neighborhood available bandwidth captures the impact of the new flow on the throughput of existing flows. These two components must be estimated separately and the combination of these two components results in the estimation of available bandwidth. In the remainder of the section, we present the definitions of these two components and the existing approaches for estimating available bandwidth.

2.1. Decomposition of available bandwidth

Since estimation of available bandwidth should be based on the priorities and contention relationships between competing flows, the value of the local achievable bandwidth and the neighborhood available bandwidth are different for different types of flows with different priorities. In this section, we define the concepts of local achievable bandwidth and neighborhood available bandwidth based on the type of flows and show how they can be combined to represent available bandwidth.

2.1.1. Decomposition of available bandwidth for realtime flows

Note that the available bandwidth of a new realtime flow is defined as the maximum transmission rate that the new flow is able to achieve without degrading the throughput of existing equal or higher priority realtime flows. This definition can be decomposed into two requirements. The first requirement is that a new realtime flow must be able to achieve its available bandwidth. Since the wireless channel is shared between competing flows, given a set of existing flows, there is a maximum transmission rate that the new realtime flow is able to achieve by competing with other flows at the MAC layer. This maximum achievable transmission rate is essentially the *local achievable bandwidth* of the new realtime flow. If this local achievable bandwidth is smaller than a new flow's throughput requirement, the new flow cannot achieve its desired throughput, indicating insufficient available bandwidth. Therefore, the available bandwidth of the new realtime flow must be no larger than this local achievable bandwidth.

The second requirement is that when a new realtime flow uses its available bandwidth, the new flow should not decrease the throughput of existing flows with equal or higher priorities. Note that enough local achievable bandwidth cannot satisfy the second requirement because a new realtime flow may achieve its local achievable bandwidth by degrading the throughput of existing flows with equal or higher priorities through MAC layer contention. To capture this, the *neighborhood available bandwidth* for a realtime flow is introduced, which is a threshold transmission rate. If the new realtime flow's transmission rate is larger than this threshold, the new flow degrades the throughput of some existing realtime flows with equal or higher priorities. The available bandwidth of a new flow must be no larger than the neighborhood available bandwidth to ensure the throughput of existing flows.

With the estimation of local achievable and neighborhood available bandwidth, the available bandwidth of a new realtime flow is essentially the minimum among local achievable bandwidth and neighborhood available bandwidth.

2.1.2. Decomposition of available bandwidth for best effort traffic

For best effort traffic, the available bandwidth is the maximum transmission rate of best effort traffic that does not degrade the throughput of

any existing realtime flows. This definition of available bandwidth essentially only includes the concept of neighborhood available bandwidth. This is because best effort flows do not have QoS requirements in terms of throughput so that there is no need for admission control on best effort flows. Therefore, the concept of local achievable bandwidth of a new best effort flow is not included in the definition of available bandwidth.

2.2. Current available bandwidth estimation approaches

Current admission control algorithms for wireless networks take one of three approaches to estimate available bandwidth. The first approach, such as VMAC [4] and SWAN [2], uses a *free bandwidth model* where idle channel time is used to estimate available bandwidth. This approach does not support priorities between flows. A best effort flow of a file transfer can occupy all of the channel bandwidth and prevent the admission of any realtime traffic. In addition, no consideration is given as to whether a new flow may affect the throughput of other neighboring flows. In the second approach [11,19,20], the *delay model*, a node uses the channel access time of its current traffic to calculate the available bandwidth of a new flow. This approach has two drawbacks. First, it does not consider the impact of admitting a new flow on other existing flows. Hence, it cannot prevent the newly admitted flow from degrading the QoS of existing flows. Second, it does not consider the fact that as a new flow is added into the network, the competition for bandwidth intensifies and the channel access time increases. Therefore, the channel access time before a node starts transmission does not reflect the bandwidth allocations after the node starts transmission. The third approach [3,15,18], the *saturation model*, estimates available bandwidth under a very conservative assumption that every active node in the network is saturated (i.e., every node always has backlogged packets). This assumption is based on an extreme state of the network where all active nodes are overloaded, which is not likely to always be true and which should be avoided to support throughput guarantees to realtime traffic. Therefore, this approach is overly pessimistic and may reduce the capacity of the network for realtime flows. To avoid the drawbacks of existing approaches, MPARC obtains accurate available bandwidth estimation by predicting local achievable bandwidth and neighborhood available bandwidth separately. Our estimation is based on our novel bandwidth allocation model that captures the bandwidth allocation between contending flows (see Sections 4–7).

3. IEEE 802.11 protocol and its extensions

To estimate the available bandwidth in networks based on IEEE 802.11, we need to understand how IEEE 802.11 and its extensions work. The IEEE 802.11 standard provides two functions in the MAC sublayer: the distributed coordination function (DCF) and the point coordination function (PCF). PCF provides contention-free frame transfer. Since PCF requires a Point Coordinator in the Access Point, it is not appropriate for a multi-hop wireless network. Hence, we only examine admission control for DCF and the extensions to DCF.

3.1. IEEE 802.11 DCF mode

IEEE 802.11 DCF provides automatic medium sharing between nodes through the use of CSMA/CA and a random backoff time following a busy medium. Prior to transfer of data packets, a node invokes the carrier-sense mechanism to determine the busy/idle state of the medium. If the medium is idle, the node defers for a constant period of time, called *DCF interframe space* (DIFS), which is determined by the physical layer. If the medium stays idle during this DIFS period, the node may transmit its packet. If the medium is busy, the node waits until the medium is observed to be idle. The length of this idle period depends on the success or failure of the previous frame. If the last frame was received correctly, the node waits DIFS time units. If the last frame was not received correctly, the node waits *extended interframe space* (EIFS) time units. After this DIFS or EIFS idle time, the node selects a random backoff period for deferring before transmitting an RTS. If the backoff timer already contains a non-zero value, the selection of a random number is not needed. The backoff period is calculated as $Backoff\ Time = Random() \times aSlotTime$, where $Random()$ is a pseudo-random integer drawn from a uniform distribution over the interval $[0, CW]$. CW, called the *contention window*, is an integer within the range of *minimum contention window* ($CW^{\min}$) and *maximum contention window* ($CW^{\max}$) (i.e., $CW^{\min} \leqslant CW \leqslant CW^{\max}$).

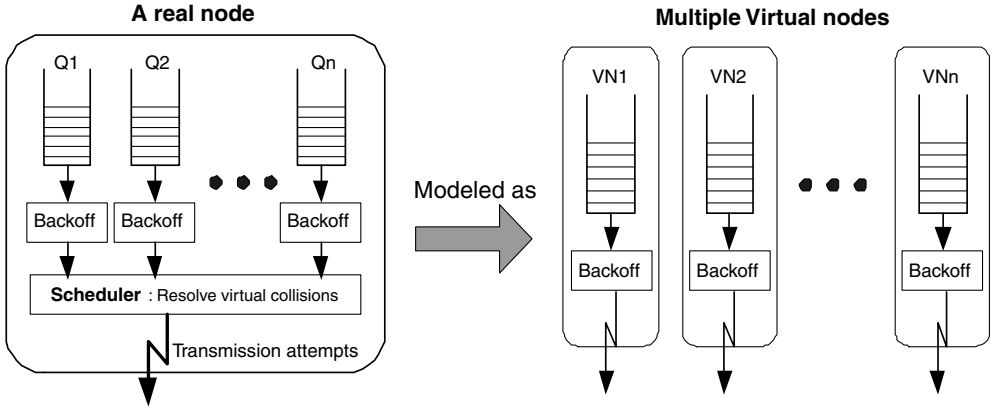


Fig. 1. Using virtual nodes to model multiple queues.

For the first transmission attempt of each packet, CW is set to $CW^{\min}$. After each unsuccessful transmission, the value of CW is doubled (*binary exponential backoff*), up to the maximum value, $CW^{\max}$. The backoff time is decremented by *aSlotTime* if the channel is idle during this period and stopped when a transmission is detected on the channel. *aSlotTime* is a constant value determined by the physical layer of the network. The backoff timer is reactivated when the channel is sensed idle again for more than DIFS time. The node transmits when the backoff timer reaches zero. At the end of every successful transmission, the CW value reverts to $CW^{\min}$ and a backoff procedure is performed immediately, even if no additional transmissions are currently queued.

The notation of the entire paper can be found in Appendix B.

3.2. Service differentiation extensions of DCF mode

In recent years, several approaches have been proposed to provide service differentiation in IEEE 802.11 by adjusting contention related parameters [1,14]. In these approaches, packets from different classes are put into different queues in a node. Each queue acts like a virtual node that observes the channel and contends for the channel independently (e.g., IEEE 802.11e [14]) as shown in Fig. 1. Therefore, in the rest of this paper, we assume that each node (which may be a virtual node) only carries traffic for a single class.

Depending on which contention related parameters are adjusted, current approaches can be separated into four categories [1,14]. First, different classes of traffic are assigned different $CW^{\min}$s. Second, different classes are assigned different packet sizes. Third, different exponential backoff schemes are used to adjust contention windows after a collision. Fourth, the DIFS is different from class to class (called AIFS in IEEE 802.11e). In [1], it is shown that the service differentiation effect of the third category is less obvious and less stable than the first two categories since it only takes effect when collisions happen, which are rare events compared to ordinary packet transmissions. Therefore, the differentiation schemes in the third category are not the focus of this paper. The schemes in the fourth category may suffer from inefficient channel usage since even if the majority of the traffic is from the class with the larger DIFS, they all must wait a very long time before they can compete for the channel. Due to this drawback, the differentiation schemes in the fourth category are again not the focus of this paper. Instead, we focus on the first and second types of methods where service differentiation is realized through different $CW^{\min}$s and frame sizes. In the next section, we present our bandwidth allocation model for networks based on this type of MAC layer.

4. Bandwidth allocation model

To estimate available bandwidth, it is necessary to understand how bandwidth is allocated between competing flows, which depends on the congestion level of the network. Since existing bandwidth allocation models [5,13] only capture saturated networks where every node is overloaded, to capture bandwidth allocation under all network load levels, we propose a novel model of bandwidth allocation. In this section, we introduce our bandwidth

allocation model in a single-hop network, where all nodes are in each other's carrier-sensing range. The extension of our model and the usage of our model for predicting available bandwidth in a multi-hop network is discussed in Section 5. Section 7 further discusses the inaccuracy caused by the extension.

To lay the groundwork for our model, we use a discrete Markov process model of the wireless channel, which is used to examine the behavior of saturated and non-saturated nodes. A *saturated node* is a node that always has backlogged packets and an *unsaturated node* is a node that is not saturated. Then, based on the states of nodes in a network, we identify three network states: saturated, non-saturated and semi-saturated. Finally, we reveal four simple equations that capture the bandwidth allocation to nodes in these three network states. For single-hop networks, these four equations enable accurate estimation of bandwidth allocation for nodes in these three network states.

4.1. Channel model

In a single-hop model, there is a fixed set $\mathcal{N} = \{1, 2, \ldots, n\}$ of transmitting nodes and every node can hear each other's transmissions since they are in each other's carrier-sensing range. Using the method derived by Bianchi [5], real time can be divided into *virtual time slots*, where a node decrements its backoff timer once per virtual time slot. Consider the example shown in Fig. 2, which represents the channel state and Node *i*'s backoff timer. Node *i*'s virtual time slots come in two types. First, a virtual time slot equals *aSlotTime* when the channel is idle (e.g., Node *i*'s first virtual time slot). However, Node *i*'s second virtual time slot extends from the beginning of the busy period until the end of the *aSlotTime* period, since the backoff timer is not decremented until after the channel becomes idle for a DIFS period. For successful transmission, there can be at most one packet sent in a virtual time slot. If multiple nodes attempt to send a packet in the same virtual time slot, a collision happens. By dividing real time into virtual time slots, the backoff process of a node can be modeled as a discrete Markov process (for details see [5]).

4.2. States of nodes

To estimate available bandwidth, it is necessary to understand the bandwidth allocation in the network, which depends on the states of the nodes. A node in a wireless network can be in two states: saturated and non-saturated. A saturated node always has backlogged packets while a non-saturated node often has an empty queue. This section presents the relationship between bandwidth allocation and node states and shows that the bandwidth share of a node depends on the states of all competing nodes in the network.

Let S_i be the amount of bandwidth allocated to a node $i \in \mathcal{N}$ and P_i be the probability that the node successfully transmits a packet in a virtual time slot. Subscript sat and $\overline{\text{sat}}$ are used to indicate saturated and non-saturated nodes respectively. For example, $S_{i,\overline{\text{sat}}}$ represents Node *i*'s bandwidth when Node *i* is a

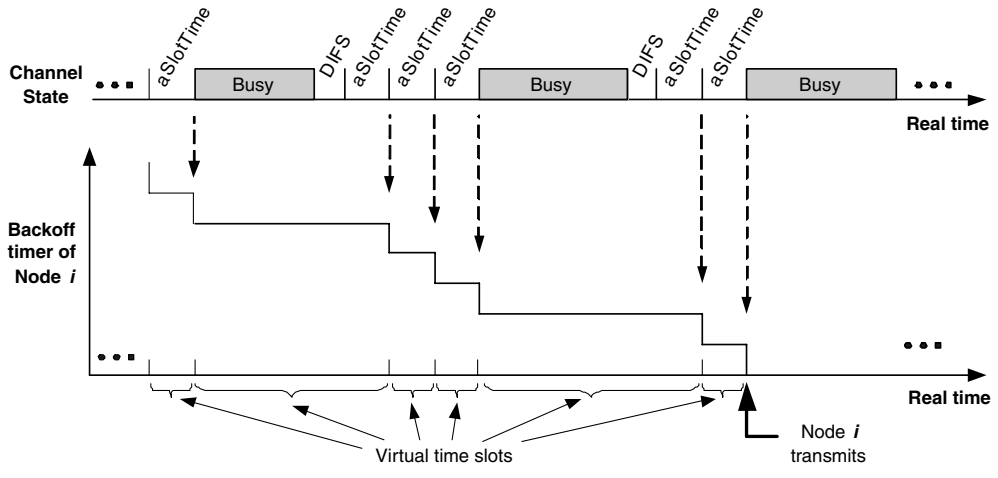


Fig. 2. Virtual time slots.

non-saturated node. W_i and L_i denote the minimum contention window size and frame size for Node i respectively, allowing our model to support the service differentiation extensions of IEEE 802.11.

The bandwidth allocated to a Node i is related to the collision probability of its packets, ϕ_i, the probability that it transmits in a randomly chosen virtual time slot, τ_i, and its load in terms of packets per second, R_i. For a saturated node, such a relationship is captured in Theorem 1 and for a unsaturated node, such a relationship is captured in Theorem 2.

Theorem 1. *For a saturated Node i,*

(1) $$P_{i,\text{sat}} = \frac{\tau_{i,\text{sat}}}{1-\tau_{i,\text{sat}}} \prod_{j=1}^{n} (1-\tau_j), \tag{1}$$

$$\tau_{i,\text{sat}} = \frac{2(1-2\phi_i)}{(1-2\phi_i)(W_i+2)+\phi_i(W_i+1)(1-(2\phi_i)^{m_i})}, \tag{2}$$

where m_i is the number of collisions that are needed for the contention window size to reach $CW^{\max}$.

(2) *$S_{i,\text{sat}}$ is the maximum bandwidth allocation of Node i and*

$$S_{i,\text{sat}} = \frac{P_{i,\text{sat}} L_i \sum_{j=1}^{n} S_j}{\sum_{j=1}^{n} P_j L_j}. \tag{3}$$

(3) *Node i is a saturated node if and only if the total amount of traffic that Node i needs to send is larger than its maximum bandwidth allocation.*

$$S_{i,\text{sat}} < R_i L_i. \tag{4}$$

Proof

(1) The proof can be found in Bianchi's work [5].
(2) Assume that in a period of real time t, there are l virtual slots. Therefore, the expected number of packets that Node i sends in t is lP_i and the throughput of Node i in period t is $S_i = lP_iL_i$. Hence,

$$\frac{S_i}{\sum_{j=1}^{n} S_j} = \frac{lP_iL_i}{\sum_{i=1}^{n} lP_jL_j} = \frac{P_iL_i}{\sum_{j=1}^{n} P_jL_j}. \tag{5}$$

By first solving for P_i in Eq. (5) and then replacing P_i with $P_{i,\text{sat}}$ and S_i with $S_{i,\text{sat}}$, we obtain Eq. (3). Since there are always packets in Node i's queue, Node i is always competing for the channel. Therefore, $S_{i,\text{sat}}$ is the maximum bandwidth allocation that Node i can achieve.

(3) If Node i is saturated (i.e., it always has packets in its queue), the packet arrival rate at Node i must be larger than its maximum bandwidth allocation. Otherwise, the queue in Node i would become empty at some time. On the other hand, if the packet arrival rate is larger than Node i's bandwidth allocation, according to queuing theory, eventually the queue length in Node $i \to \infty$. Therefore, Node i is saturated. □

Theorem 2. *For any non-saturated Node i,*

$$S_{i,\overline{\text{sat}}} = R_i L_i, \tag{6}$$

$$S_{i,\overline{\text{sat}}} \leqslant S_{i,\text{sat}}, \tag{7}$$

$$P_{i,\overline{\text{sat}}} = \frac{R_i \sum_{j=1}^{n} P_j L_j}{\sum_{j=1}^{n} S_j}, \tag{8}$$

$$P_{i,\overline{\text{sat}}} < P_{i,\text{sat}}. \tag{9}$$

Proof

(1) Since Node i is unsaturated, its queue is often empty. Therefore, when Node i finishes transmitting, it often has no more packets in its queue to transmit. This happens when the total amount of traffic that Node i needs to transmit is smaller than the maximum throughput (see Theorem 1) that Node i can achieve (Inequality (7)). In this case, the node's throughput is the same as its load (Eq. (6)).
(2) Combining Eq. (5) and Eq. (6) results in Eq. (8).
(3) Because the node often has no packet to transmit during idle periods, it is obvious that the probability that it transmits in a virtual time slot is lower than the case when the node always has backlogged packets and is always trying to transmit (Inequality (9)). □

Essentially, Theorem 1 shows that the maximum bandwidth allocation to a saturated node is constrained by its W_i and ϕ_i and Theorem 2 demonstrates that for a non-saturated node, the limiting factor of its bandwidth allocation is actually its load R_i. As can be seen from Eqs. (3) and (6), the bandwidth allocation to a saturated node depends on both the node's own state and the bandwidth allocations of the other nodes, which in turn is related to

the states of the other nodes. Therefore, the bandwidth allocation to a node is related to the congestion level of the whole network.

4.3. States of networks

Since bandwidth allocation to a flow is related to the congestion level of the whole network, it is important to identify different congestion levels and understand how they affects bandwidth allocation. The congestion levels of a network can be classified into three states: saturated, non-saturated or semi-saturated. A network is in a *saturated state* when every node always has backlogged packets. In a *non-saturated network*, every node is non-saturated, indicating a lightly loaded network. A *semi-saturated* network is between the saturated and non-saturated state, where some of the nodes are saturated while other nodes are non-saturated.

To better illustrate the relationship between bandwidth allocation and network state, we present a simple NS2 [9] simulation using the topology shown in Fig. 3. The channel capacity of the network is 2 Mbps. The queue size in each node is 50 packets. The packet size is 512 bytes. The simulation runs for 150 s. There are four nodes in the network with Nodes A and C transmitting to Nodes B and D, respectively.

Figs. 4 and 5 depict the queue length and the throughput of Nodes A and C. From time 5 s to time 50 s, Nodes A and C each carry a realtime flow that generates 50 packets per second. The queues in both nodes are often empty during this period, indicating a non-saturated network. Both flows can achieve throughput that matches their packet generation rates. At time 50 s, the traffic type of Node A changes to a file transfer. The queue in Node A becomes full while the queue in Node C is still often empty, indicating a semi-saturated network. During this period, even though Node A tries to send more packets, it is not able to "push down" Node C's

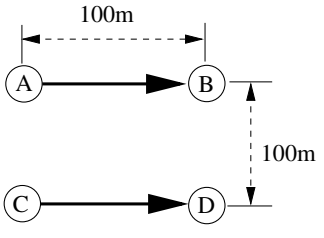


Fig. 3. Topology.

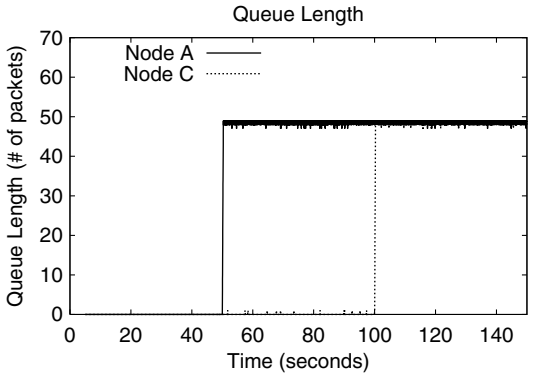


Fig. 4. Queue length of Nodes A and C.

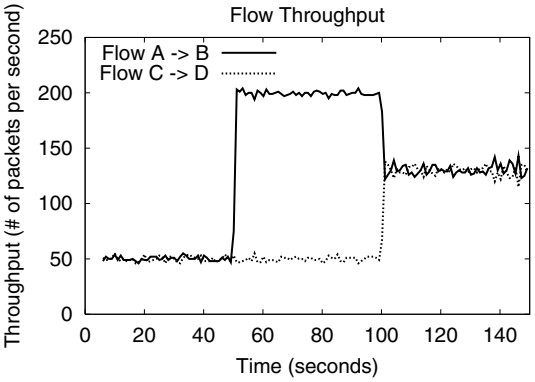


Fig. 5. Throughput of Nodes A and C.

bandwidth allocation. From time 100 s to time 150 s, the realtime traffic in Node C increases its generating rate to 300 packets per second. Both queues in Node A and Node C become constantly full, indicating a saturated network. During this period, Nodes A and C share the channel bandwidth equally and the realtime traffic in Node C is unable to achieve its desired bandwidth.

This example shows that bandwidth allocations are related to the state of the network. Depending on the traffic load and type, a practical network can be in any of the three states. Therefore, to estimate available bandwidth to a flow, we must capture bandwidth allocation in all network states.

4.4. Bandwidth allocation for different networks states

Due to the relationship between bandwidth allocation and network states, in this section, we analyze the bandwidth allocation for each of the three network states: saturated, unsaturated and

semi-saturated. Our analysis reveals four very simple mathematical equations (Eqs. (10) and (23)–(25)), which can be used to calculate the bandwidth allocated to each node under all network states. In Sections 5 and 6, these results are used by MPARC to estimate available bandwidth, so that admission control and rate policing can be performed.

4.4.1. Semi-saturated network

Consider a semi-saturated network, where the set of saturated nodes is N_1, the set of unsaturated nodes is N_2 and $N_1 \cup N_2 = \mathcal{N}$. To solve the S_i for any Node i in such a network, it is necessary to determine the state of Node i. Note that Theorems 1 and 2 show that Node i's bandwidth allocation S_i has an upper bound determined by $S_{i,\text{sat}}$ and R_iL_i. If $S_{i,\text{sat}}$ is larger than its load R_iL_i, Node i is non-saturated and its bandwidth allocation equals R_iL_i. If $S_{i,\text{sat}}$ is smaller than R_iL_i, Node i is saturated and its bandwidth allocation becomes $S_{i,\text{sat}}$. Therefore, as long as $S_{i,\text{sat}}$ is known, the bandwidth allocation of Node i can be easily determined according to the offered load on Node i.

To calculate $S_{i,\text{sat}}$, in Theorem 3, we identify the relationship between $S_{i,\text{sat}}$ and the network load.

Theorem 3. *Given the maximum throughput of the channel as C, the maximum bandwidth allocated to Node i can be approximated as*

$$S_{i,\text{sat}} \approx \frac{L_i C}{\eta W_i}, \tag{10}$$

where

$$\eta = \frac{\sum_{i\in N_1} \frac{L_i}{W_i}}{1 - \sum_{i\in N_2} \frac{R_i L_i}{C}}. \tag{11}$$

Proof. Since the saturated nodes in the network always have packets to transmit and hence fill up the network bandwidth,

$$\sum_{i=1}^{n} S_i \approx C. \tag{12}$$

Combining Eqs. (1) and (3) with Eq. (12)

$$S_{i,\text{sat}} = \frac{\frac{\tau_{i,\text{sat}}}{1-\tau_{i,\text{sat}}} \prod_{j=1}^{n}(1-\tau_j) L_i C}{\sum_{j=1}^{n} P_j L_j}. \tag{13}$$

It is obvious that $S_{i,\text{sat}}$ depends on $\sum_{j=1}^{n} P_j L_j$, which is the average number of bits transmitted in a virtual time slot. By defining $\rho = \sum_{i=1}^{n} P_i L_i$, $S_{i,\text{sat}}$ can be expressed as

$$S_{i,\text{sat}} = \frac{\frac{\tau_{i,\text{sat}}}{1-\tau_{i,\text{sat}}} \prod_{j=1}^{n}(1-\tau_j) L_i C}{\rho}, \tag{14}$$

where $\tau_{i,\text{sat}}$ depends on W_i and ϕ_i (Eq. (2)). In [13,22], it has been proved that when $W_i \gg 1$ (e.g., $W_i > 10$), which is often true in real IEEE 802.11 wireless networks, ϕ_i is roughly the same for all nodes. Therefore,

$$\phi_i \approx \phi_j = \phi \quad \text{for some constant } \phi. \tag{15}$$

Combining Eq. (2) with Eq. (15) and using the fact that $W_i \gg 1$ and $m_i \approx m_j = m$, the following approximation can be made:

$$\frac{1-\tau_{i,\text{sat}}}{\tau_{i,\text{sat}}} \approx \frac{1}{2}\left(1 + \phi\frac{1-(2\phi)^m}{1-2\phi}\right) W_i = f(m,\phi) W_i, \tag{16}$$

where $f(m,\phi) = \frac{1}{2}(1 + \phi\frac{1-(2\phi)^m}{1-2\phi})$. Combining Eq. (16) with Eq. (14),

$$S_{i,\text{sat}} = \frac{\prod_{j=1}^{n}(1-\tau_j) L_i C}{f(m,\phi) W_i \rho}. \tag{17}$$

Combining Eqs. (1), (2), (8) and (12), ρ satisfies the following relationship:

$$\rho = \sum_{i=1}^{n} P_i L_i = \sum_{i\in N_1} P_{i,\text{sat}} L_i + \sum_{i\in N_2} P_{i,\overline{\text{sat}}} L_i \tag{18}$$

$$= \sum_{i\in N_1} \frac{\tau_{i,\text{sat}} L_i}{1-\tau_{i,\text{sat}}} \prod_{j=1}^{n}(1-\tau_j) + \sum_{i\in N_2} \frac{R_i L_i \rho}{C}. \tag{19}$$

Solving for ρ in Eq. (19),

$$\rho = \frac{\sum_{i\in N_1} \frac{\tau_{i,\text{sat}}}{1-\tau_{i,\text{sat}}} L_i \prod_{j=1}^{n}(1-\tau_j)}{1 - \sum_{i\in N_2} \frac{R_i L_i}{C}}. \tag{20}$$

Applying the approximation in Eq. (16)–(20) results in

$$\rho \approx \frac{\sum_{i\in N_1} \frac{L_i}{W_i} \prod_{j=1}^{n}(1-\tau_j)}{(1 - \sum_{i\in N_2} \frac{R_i L_i}{C}) f(m,\phi)} = \eta \frac{\prod_{j=1}^{n}(1-\tau_j)}{f(m,\phi)}, \tag{21}$$

where $\eta = \frac{\sum_{i\in N_1} \frac{L_i}{W_i}}{1-\sum_{i\in N_2} \frac{R_i L_i}{C}}$.

Combining Eqs. (21) and (17), $S_{i,\text{sat}}$ in a semi-saturated network can be expressed as

$$S_{i,\text{sat}} = \frac{L_i C}{\eta W_i}. \qquad \square \tag{22}$$

Theorem 3 shows that to calculate $S_{i,\text{sat}}$, we must calculate η, which represents the network load and depends on the states of all nodes in the network. To calculate η, note that Eq. (10) shows that the maximum bandwidth allocation ($S_{i,\text{sat}}$) to a Node i

is determined by the η of the whole network. The larger the η, the smaller the $S_{i,\mathrm{sat}}$. According to Theorem 1, when $R_iL_i > S_{i,\mathrm{sat}}$, Node i becomes saturated. Therefore, as η increases, $S_{i,\mathrm{sat}}$ decreases so that more and more nodes in the network become saturated. When Node i is at the edge of turning from non-saturated to saturated, $R_iL_i = S_{i,\mathrm{sat}}$. Combined with Eq. (10), the threshold value of η at this turning point, η_i^*, can be expressed as

$$\eta_i^* = \frac{C}{R_i W_i}. \tag{23}$$

Sorting the nodes according to their η_i^* in ascending order results in a sequence of nodes $(x_1, x_2, \ldots, x_n)$ where $\eta_{x_i}^* \leqslant \eta_{x_j}^*$ if $i < j$. If $\eta_{x_k}^* < \eta < \eta_{x_{k+1}}^*$, nodes $x_1, \ldots, x_k$ are saturated and nodes $x_{k+1}, \ldots, x_n$ are non-saturated. Therefore,

$$\eta = \eta(k) = \frac{\sum_{i=1}^{k} \frac{L_{x_i}}{W_{x_i}}}{1 - \sum_{i=k+1}^{n} \frac{R_{x_i} L_{x_i}}{C}}, \tag{24}$$

$$\eta_{x_k}^* \leqslant \eta(k) < \eta_{x_{k+1}}^*. \tag{25}$$

Since the range of k is the number of competing neighboring nodes, which is generally not large, we can calculate the value of η corresponding to each value of k using Eq. (24). As shown in Appendix A, it can be proved that there is only one valid combination of (η, k) that can satisfy both Eq. (24) and Inequality (25), which is essentially a valid solution to η and k in the network. With the solution (η, k), the state of the nodes can be decided, where the saturated nodes are $N_1 = \{x_1, x_2, \ldots, x_k\}$ and the non-saturated nodes are $N_2 = \{x_{k+1}, x_{k+2}, \ldots, x_n\}$. The bandwidth allocation to every node, hence, can be determined as

$$S_i = \begin{cases} \frac{L_i C}{\eta W_i}, & i \in N_1, \\ R_i L_i, & i \in N_2. \end{cases} \tag{26}$$

4.4.2. Saturated networks

Note that in deriving Eq. (26), we assume that the network is semi-saturated, meaning that both N_1 and N_2 are non-empty. However, it is also possible that the network is saturated or unsaturated.

For a saturated network, since all nodes are saturated, $\eta > \eta_j^*$ for any Node j in the network. By setting $\eta_{x_{n+1}}^* = \infty$, the solution of η satisfying Eq. (24) and Inequality (25) is obtained at $k = n$ for a saturated network, where $\eta_{x_{n+1}}^* > \eta(n) = \sum_{i=1}^{n} \frac{L_{x_i}}{W_{x_i}} > \eta_{x_n}^*$. In this case, N_2 is empty and it is easy to check that Eq. (26) is still valid for calculating S_i, although only the part corresponding to N_1 is used.

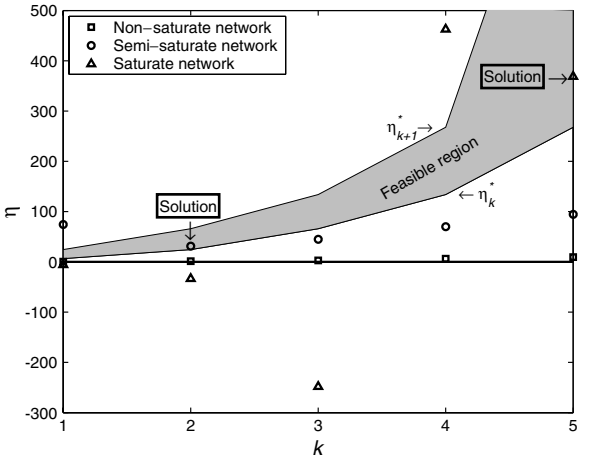


Fig. 6. Example of η, η^* and the corresponding solution.

4.4.3. Non-saturated networks

For a unsaturated network, since none of the nodes in the network are saturated, $\eta < \eta_j^*$ for any Node j in the network. Therefore, there is no solution of η satisfying Eq. (24) and Inequality (25) for all $1 \leqslant k \leqslant n$. By setting $\eta_{x_0}^* = 0$, $k = 0$ becomes the solution. In such case, N_1 is empty and Eq. (26) is still valid for calculating S_i, although only the part corresponding to N_2 is used.

To illustrate how to solve Eq. (24) and Inequality (25), Fig. 6 shows an example of $\eta(k)$ in a five node network in saturated, non-saturated and semi-saturated states, respectively. The points in the figure represent the values of η corresponding to k calculated using Eq. (24). The inequality constraint (25) is represented by the shaded area. When a point for η is located in the shaded area, the point represents a valid solution for η. In Fig. 6, the solution for a saturated network is achieved when $k = 5$, the solution for a semi-saturated network is achieved when $k = 2$, and the non-saturated network has no solution for $1 \leqslant k \leqslant 5$.

Since η and k for all network states can be solved, the bandwidth allocation to every node can be determined using Eq. (26). With the knowledge of bandwidth allocation, predictions of available bandwidth to a flow can be achieved.

5. Prediction of available bandwidth

Two major parts are needed to use the bandwidth allocation model in Section 4 to perform admission and rate policing in MPARC. The first part describes the methods for estimating the available

bandwidth of a flow, which is the focus of this section. The second part describes the signaling process, which focuses on how to collect information that is needed for available bandwidth estimation and how admission control and rate policing is performed based on estimated available bandwidth of a flow. Section 6 explains the details of the second part.

To predict available bandwidth in multi-hop ad hoc networks, we make a simple extension to our bandwidth allocation model for a single-hop network. In this extension, the bandwidth allocation between Node k and its contending neighbors is approximated as if Node k and its contending neighbors constitute a single-hop network. As an approximation, this extension introduces inaccuracy in estimating available bandwidth in a multi-hop network. Hence, in Sections 7 and 8, we discuss the sources and impact of the inaccuracy on our prediction model and other existing methods.

The prediction of available bandwidth is important for both admission control for realtime traffic and rate policing for best effort traffic. A realtime flow can only be admitted at a node if there is enough available bandwidth for the new flow. The transmission rate of best effort traffic at a node should be adjusted to be no more than the available bandwidth of best effort traffic. To estimate the available bandwidth, it is important to use our bandwidth allocation model to capture the effect of network contentions. Hence, in the remainder of the section, we discuss how to estimate local achievable bandwidth and neighborhood available bandwidth, which are the two components for available bandwidth (see Section 2.1) using our bandwidth allocation model. The combination of the estimated local achievable bandwidth and neighborhood available bandwidth produces the estimated available bandwidth. The other existing methods for estimating available bandwidth and their limitations are discussed in Section 8.

5.1. Prediction of local achievable bandwidth

To predict the available bandwidth for a realtime flow, both local achievable bandwidth and neighborhood available bandwidth need to be predicted. In this section, we discuss how to predict local achievable bandwidth and in Section 5.2, we show how to predict neighborhood available bandwidth.

Since multiple nodes on the route of a flow f may contend with each other, the local achievable bandwidth for flow f is affected by this intra-flow contentions. For example, if flow f has two hops as shown in Fig. 7, both Nodes A and B are in each other's carrier-sensing range and only one node can transmit at a time. Therefore, if flow f transmits at rate R_f with packet size L_f, both Nodes A and B experience a traffic load of $2R_f L_f$ on the channel. As R_f or L_f increase, the first node among Nodes A and B that turns saturated becomes the bottleneck of the flow. After the bottleneck node becomes saturated, even if flow f increases its transmission rate, the throughput of the flow does not increase. Hence, the saturation throughput of the bottleneck node essentially determines the local achievable bandwidth of flow f, denoted as U_f^l.

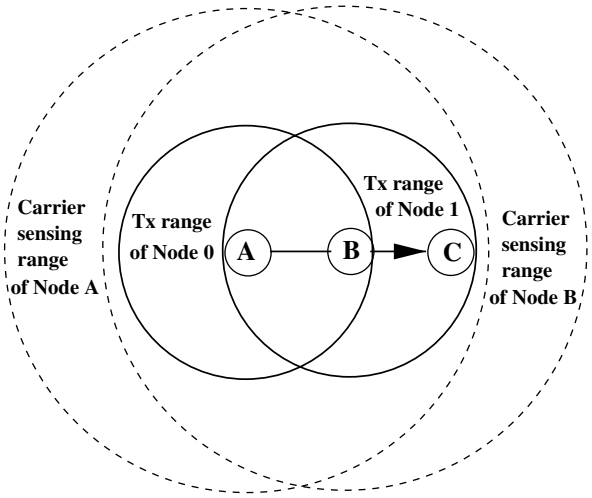


Fig. 7. Multi-hop flows.

In general, if Node k is the bottleneck node for a multi-hop flow f and flow f just achieves its local achievable bandwidth, Node k is at the edge between saturated and unsaturated. Therefore, based on Eq. (10), the local achievable bandwidth of flow f, U_f^l, is

$$U_f^l = R_f^l L_f = S_{k,\mathrm{sat}} = \frac{CL_f}{\eta W_k}, \tag{27}$$

where R_f^l is the transmission rate of flow f when it just achieves its local achievable bandwidth.

To calculate U_f^l, we next show how to get the value of η when flow f is transmitting at rate R_f^l. Assume there are α nodes (including Node k itself) along the route of flow f that are also in Node k's carrier-sensing range. Except Node k, the other $(\alpha - 1)$ nodes are unsaturated when the flow is sending at rate R_f^l. Hence, using Eq. (24), the η at Node k is

$$\eta = \frac{\frac{L_f}{W_k} + \sum_{i\in N_1} \frac{L_i}{W_i}}{1 - \sum_{i\in N_2} \frac{R_i L_i}{C} - (\alpha - 1)\frac{R_f^1 L_f}{C}}, \tag{28}$$

where $\frac{L_f}{W_k}$ represents the saturated Node k's load on the network while $(\alpha-1)\frac{R_f^1 L_f}{C}$ captures the load of the $(\alpha-1)$ unsaturated nodes on the route of the flow that are also competing with Node k. Since when the flow rate is R_f^1, Node k is at the edge of turning from unsaturated to saturated, $\eta = \eta_k^*$. Combining with Eq. (23) results in

$$R_f^1 = \frac{C}{\eta_k^* W_k} = \frac{C}{\eta W_k}. \tag{29}$$

By replacing R_f^1 in Eq. (28) using Eq. (29) and then solving for η

$$\eta = \frac{\alpha\frac{L_f}{W_k} + \sum_{j\in N_1} \frac{L_j}{W_j}}{1 - \sum_{i\in N_2} \frac{R_i L_i}{C}}, \tag{30}$$

$$\eta \geqslant \eta_j^* \quad \forall j \in N_1, \tag{31}$$

$$\eta < \eta_i^* \quad \forall i \in N_2. \tag{32}$$

Note that Eq. (30) and Inequalities (31) and (32) define a piece-wise linear function between $\alpha\frac{L_f}{W_k}$ and η. Given the loads on Node k's neighbors, Node k is able to build such a piece-wise linear function. Given α, L_f and W_k, Node k can immediately determine the corresponding η and use this η to find its local achievable bandwidth U_f^1 according to Eq. (27).

An example of the piece-wise linear function between $\alpha\frac{L_f}{W_k}$ and η is shown in Fig. 8, where Node k has five competing neighbors. The piece-wise function consists of five line segments and the ends of the line segments correspond to the η_i^*s of the five competing neighbors.

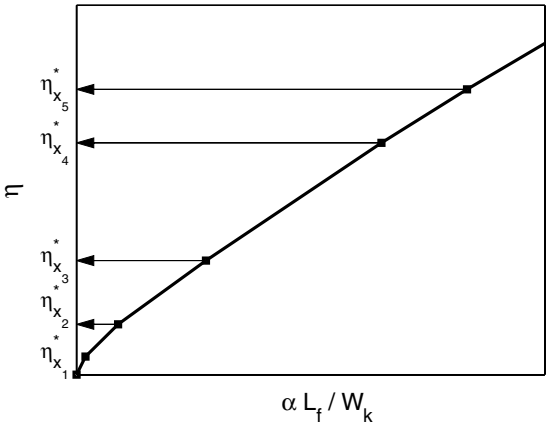


Fig. 8. Example of the piece-wise linear relationship between η and $\alpha\frac{L_f}{W_k}$. In the figure, $\eta_{x_i}^* \leqslant \eta_{x_j}^*$ if $i < j$.

The algorithm for building the piece-wise function and predicting the local achievable bandwidth is presented in Algorithm 1. For a Node k, given α, neighboring nodes' contention window allocation, **W**, their packet size, **L**, their load, **R** and the network capacity C, the U_f^1 can be calculated immediately by this algorithm.

Algorithm 1. Prediction of local achievable bandwidth of Node k

PREDICTION OF U_f^1 $(C, \mathbf{R}, \mathbf{L}, \mathbf{W}, \alpha, k)$
/* Calculating η^* */
for $i = 1$ to n **do**
$\eta_i^* \leftarrow \frac{C}{W_i R_i}$
end for
SORT $(\mathbf{R}, \eta^*)$; SORT $(\mathbf{L}, \eta^*)$; SORT $(\mathbf{W}, \eta^*)$
$V_f \leftarrow \alpha\frac{L_f}{W_k}$
/* Calculating η corresponding to $V_f = \alpha\frac{L_f}{W_k}$ */
$X_0 \leftarrow 0$
$Y_0 \leftarrow \sum_{i=1}^{n} \frac{R_i L_i}{C}$
for $i = 1$ to n **do**
$X_i \leftarrow X_{i-1} + \frac{L_i}{W_i}$
$Y_i \leftarrow Y_{i-1} - \frac{R_i L_i}{C}$
$V_{f,i}^* \leftarrow \eta_i^*(1 - Y_i) - X_i$
if $V_{f,i-1}^* \leqslant V_f < V_{f,i}^*$ **then**
$\eta \leftarrow \frac{X_{i-1}+V_f}{1-Y_{i-1}}$; BREAK
end if
end for
/* Calculating U_f^1 */
$U_f^1 \leftarrow \frac{V_f}{\alpha\eta} C$;

SORT$(array, index)$
Sort *array* in ascending order of *index*

5.2. Neighborhood available bandwidth prediction

The prediction of neighborhood available bandwidth is needed by both admission control and rate policing. Admission control needs to predict neighborhood available bandwidth to determine if a new realtime flow will affect the throughput of other existing flows. Rate policing needs to know the neighborhood available bandwidth to determine the maximum allowable sending rate for best effort flows.

Since the neighborhood available bandwidth of a new flow is related to the priority of the new flow, different flows have different neighborhood available bandwidth. For example, a best effort flow is not allowed to reduce the bandwidth of existing realtime flows while a high priority realtime flow is allowed to degrade the bandwidth of both lower priority realtime flows and best effort flows. Therefore, the neighborhood available bandwidth of a best effort flow can be much smaller than the neighborhood available bandwidth of a high priority realtime flow.

To calculate Flow f's neighborhood available bandwidth, U_f^{n}, at a Node k, note that

$$U_f^{\mathrm{n}} = R_f^{\mathrm{n}} L_f, \tag{33}$$

where R_f^{n} is the throughput of flow f when it achieves the neighborhood available bandwidth. When the rate of the flow is R_f^{n}, the η at Node k can be expressed as

$$\eta = \frac{\sum_{j\in N_1}\frac{L_j}{W_j}}{1-\sum_{i\in N_2}\frac{R_i L_i}{C}-\alpha\frac{R_f^{\mathrm{n}} L_f}{C}}, \tag{34}$$

$$\eta \geqslant \eta_j^* \quad \forall j \in N_1, \tag{35}$$

$$\eta < \eta_i^* \quad \forall i \in N_2. \tag{36}$$

Combining Eqs. (33)–(36),

$$\alpha U_f^{\mathrm{n}} = C \times \begin{cases} 1-\sum\limits_{i\in N_2}\frac{R_i L_i}{C}-\frac{1}{\eta}\sum\limits_{j\in N_1}\frac{L_j}{W_j}, \\ \quad \text{for } \eta_j^* \leqslant \eta < \eta_i^* \ \forall j \in N_1 \ \forall i \in N_2, \\ 1-\sum\limits_{i\in \mathscr{N}}\frac{R_i L_i}{C}, \\ \quad \text{for } 0 \leqslant \eta < \eta_i^* \ \forall i \in \mathscr{N}, \\ 1-\frac{1}{\eta}\sum\limits_{i\in \mathscr{N}}\frac{L_i}{W_i}, \\ \quad \text{for } \eta_i^* \leqslant \eta \ \forall i \in \mathscr{N}. \end{cases} \tag{37}$$

Eq. (37) defines a piece-wise linear function between αU_f^{n} and $1/\eta$, which can be pre-calculated and stored in a node. An example of the bandwidth prediction function is shown in Fig. 9, where there are five competing nodes. The bandwidth prediction function consists of six line segments. The five end points of these line segments correspond to the $1/\eta_i^*$s of the five competing nodes, which can be easily calculated based on the traffic load information and Eq. (23). It can be seen that a larger αU_f^{n} corresponds to smaller $1/\eta$ in the bandwidth prediction function. As $1/\eta$ becomes smaller than the reciprocal of a Node i's saturation threshold $1/\eta_i^*$, Node i is pushed to its saturated state by the new flow and the throughput of Node i's flows decreases. Based on this observation, the neighborhood available bandwidth for flow f can be calculated as follows.

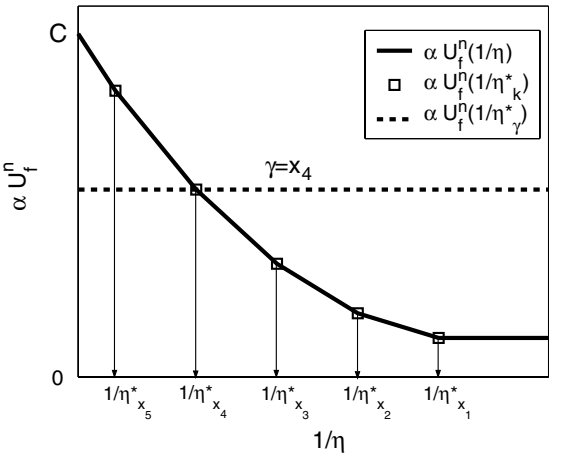


Fig. 9. Piece-wise linear function of αU_f^{n} and $1/\eta$.

Note that if flow f is a realtime flow, it should not degrade the throughput of any existing realtime flows with equal or higher priorities. Therefore, if Node γ is the node with the lowest saturation threshold among all the nodes that carry traffic with equal or higher priority than flow f, based on Eq. (37), the neighborhood available bandwidth of flow f can be expressed as

$$U_f^{\mathrm{n}} = \frac{C}{\alpha}\left(1-\sum_{i\in N_2}\frac{R_i L_i}{C}-\frac{1}{\eta_\gamma^*}\sum_{j\in N_1}\frac{L_j}{W_j}\right), \tag{38}$$

where

$$\eta_j^* \leqslant \eta_\gamma^* \quad \forall j \in N_1, \tag{39}$$

$$\eta_i^* > \eta_\gamma^* \quad \forall i \in N_2. \tag{40}$$

For example, in Fig. 9, the dotted line represents αU_f^{n} of flow f when Node γ is Node x_4.

If flow f is a best effort flow, it should not degrade the throughput of any realtime flow. Hence, assuming that Node λ is the node with the lowest saturation threshold among all the nodes that carry realtime traffic, the neighborhood available bandwidth of best effort traffic is

$$U_f^{\mathrm{n}} = \frac{C}{\alpha}\left(1-\sum_{i\in N_2}\frac{R_i L_i}{C}-\frac{1}{\eta_\lambda^*}\sum_{j\in N_1}\frac{L_j}{W_j}\right), \tag{41}$$

where

$$\eta_j^* \leqslant \eta_\lambda^* \quad \forall j \in N_1, \tag{42}$$

$$\eta_i^* > \eta_\lambda^* \quad \forall i \in N_2. \tag{43}$$

Algorithm 2 shows the pseudocode for calculating neighborhood available bandwidth. For a Node k, given α, γ, neighboring nodes' contention window allocation, **W**, their packet size, **L**, their load, **R** and the network's capacity, C, the U_f^{n} can be calculated immediately by this algorithm.

As discussed in Section 2.1, combining U_f^{l} (Eq. (27)) and U_f^{n} (Eq. (38)), the available bandwidth of a realtime flow f can be finally expressed as $\min(U_f^{\mathrm{l}}, U_f^{\mathrm{n}})$. The available bandwidth of a best effort flow is simply U_f^{n} (Eq. (41)).

Algorithm 2. prediction of neighborhood available bandwidth of Node k

PREDICTION OF $U_f^{\mathrm{n}}(C, \mathbf{R}, \mathbf{L}, \mathbf{W}, \alpha, \gamma)$

$\eta_\gamma^* \leftarrow \frac{C}{W_\gamma R_\gamma}$

$X \leftarrow \sum_{j:\eta_j^* \leqslant \eta_\gamma^*} \frac{L_j}{W_j}$

$Y \leftarrow \sum_{i:\eta_i^* > \eta_\gamma^*} \frac{R_i L_i}{C}$

$U_f^{\mathrm{n}} \leftarrow \frac{C}{\alpha}[(1-Y) - \frac{1}{\eta_\gamma^*} X]$

6. Signaling process for admission control and rate policing

To use the available bandwidth estimation method described in Section 5, there are two remaining issues that need to be solved. First, the necessary information for our available bandwidth estimation method need to be collected. Second, we need to know how to perform admission control and rate policing based on the estimated available bandwidth. In this section, we address these two issues.

6.1. Collection of neighbor information

The analysis in Section 5 shows that a flow's available bandwidth is related to the loads and traffic classes of the nodes in its carrier-sensing range. Therefore, to predict the available bandwidth of a flow, it is necessary to collect traffic information at a node's competing neighbors, which includes reservations and classes of realtime traffic and the average packet arrival rate and size of best effort traffic. Since a node contends for bandwidth not only with its neighbors in its transmission range, but also with its near-neighbors in carrier-sensing range, the node must collect multi-hop neighbors' traffic information. In our experiments, we use three hops as the collection range. This is purely a heuristic and does not guarantee to involve all contending nodes and may involve non-contending nodes. More elaborate methods, such as using the locations of nodes to decide contention relationships, may be used to improve the accuracy of finding contending nodes.

In MPARC, every node i periodically broadcasts its traffic information (W_i, L_i, R_i as well as flow priority) in its one-hop neighborhood. The broadcast message also carries traffic information of its two-hop neighbors, which is gathered through listening to other nodes' broadcasts. Using this method, every node learns the traffic of competing nodes in its three-hop neighborhood. Besides periodic updates, an update can also be triggered when a new reservation is made. The message overhead of update messages can be reduced by piggybacking load information on control and data packets, adding minimal overhead to heavily loaded networks.

By monitoring the traffic information broadcasted in its neighborhood, a Node k can gather the addresses of its three-hop neighbors, **A**, their contention window allocation, **W**, their packet size, **L**, their traffic load, **R**, and their traffic priority. Given the path $\mathscr{P}$ of a flow, Node k can also calculate the α of the flow as the number of common nodes between the set of Node k's 3-hop neighboring nodes and the path $\mathscr{P}$, which can be expressed as

$$\alpha = |\mathscr{P} \cap A|. \tag{44}$$

Hence, by collecting traffic information of neighboring nodes, Node k can gain enough knowledge to calculate the available bandwidth of a flow using the method described in Section 5.

6.2. Admission control and rate policing

Given the estimation of available bandwidth, MPARC performs admission control and rate policing as follows. For realtime flows, before admission control is performed, some ad hoc routing protocol (e.g., DSR [10], DSDV [17] or AODV [16]) has been used to find the route for a new flow. Then, a QoS signaling protocol, such INSIGNIA [12] or RSVP [6], is used to set up admission control and resource reservation at each node along the route. In brief, a reservation request message, which carries the flow route, packet length, traffic class and flow rate information, is sent along the route of the new flow. Each node that receives this message performs admission control using the estimated

available bandwidth. If admission control succeeds, a soft bandwidth reservation is made and the reservation request message is forwarded to the next hop. If admission control succeeds at every node, this route has enough bandwidth for the new flow. Therefore, the destination can send back a reservation confirmation message to the source of the flow so that the new flow can start. If admission control fails at some node, the flow is rejected and the reservation is torn down using explicit messages or timeouts. For rate policing of best effort flows, no resource reservation is needed. A node with best effort traffic estimates the available bandwidth for best effort traffic and uses a rate control mechanism, such as a leaky bucket, to regulate its transmission rate of best effort packets below the available bandwidth.

7. Inaccuracy of bandwidth prediction in multi-hop networks

The accuracy of our bandwidth allocation model in Section 4 depends on the assumption that every node competes with the same set of nodes. However, when we extend the model to predict available bandwidth in a multi-hop network, such an assumption breaks in two cases. In the first case, the neighbors of Node i may compete with other nodes that are not in Node i's carrier-sensing range. In the second case, the contenting neighbors of a Node i may not be in each other's carrier-sensing range. Both cases can affect the accuracy of our prediction method as well as all other existing bandwidth estimation methods. In the remainder of this section, we discuss the degree of impact of these two cases on our bandwidth prediction model and other existing methods. In Section 9, we show that although our model cannot accurately capture contention between nodes in multi-hop networks, it still gives much better predictions of available bandwidth than existing methods due to the reasons discussed in Section 8.

7.1. Effects of different neighborhood

Since neighbors of Node i may compete with other nodes that are not in Node i's carrier-sensing range, the assumption that each neighboring node of Node i sees the same channel state breaks. For example, in Fig. 10, Node B is in the carrier-sensing range of both Nodes C and A. Therefore, when Node A is transmitting, Node B does not compete with Node C. This case is not considered in our single-hop bandwidth allocation model, therefore, inaccuracy is introduced in the available bandwidth prediction at Node C. However, we believe that this inaccuracy is very small. From Node C's prospective, when Node B is blocked by Node A's transmission, Node B's behavior is similar to when Node B does not have any packets for transmission at that time. Therefore, to Node C, Node B is equivalent to an unsaturated node and can be captured in our bandwidth allocation model. Hence, in this case, although the underlying assumption of our bandwidth allocation model may not hold, the available bandwidth estimation provided by our model is still highly accurate.

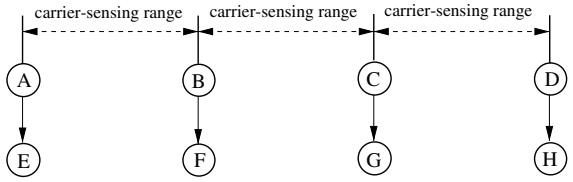


Fig. 10. Different neighbor sets.

For the delay and free bandwidth models, when the neighbors of Node i compete with other nodes that are not in Node i's carrier-sensing range, the prediction accuracy is also not degraded significantly. Using the same example in Fig. 10, Node C's measurements of delay or free bandwidth do not differ whether Node B is an unsaturated node or Node B is blocked by Node A's transmission. Hence, the prediction accuracy of the delay and free bandwidth models is not further degraded in multi-hop networks due to this case.

However, the saturation model's prediction accuracy may be affected significantly. In Fig. 10, since Node C assumes that Node B is saturated, Node C does not consider Node A's blocking effects on Node B. Therefore, Node C's prediction of its available bandwidth may be much smaller than the actual value.

7.2. Effects of non-competing neighbors

When the contenting neighbors of a Node i are not in each other's carrier-sensing range, they may transmit concurrently. Such concurrent transmissions are not captured in our bandwidth prediction model and result in inaccurate bandwidth prediction at Node i. For example, in Fig. 10, although both Nodes B and D compete for bandwidth with

Node C, Nodes B and D cannot hear each other and hence can transmit concurrently. However, when Node C uses our bandwidth allocation model to predict achievable bandwidth, Node C assumes that Nodes B and D can hear each other and compete with each other for bandwidth. This false assumption makes Node C overestimate the network's congestion level, which results in a underestimation of the available bandwidth at Node C. Such over-pessimistic estimation of available bandwidth may result in conservative admission control and rate policing.

Similarly, the saturation model also omits the effects of concurrent transmissions between Nodes B and D and its estimation accuracy is also degraded due to this effect. This case also increases the prediction inaccuracy of the delay and free bandwidth models. This is due to the fact that before Node C starts its traffic, Nodes B and D are not likely to be synchronized in their transmissions. Therefore, the chances that their transmissions happen concurrently may be very small. For example, assume that the channel capacity is 5 packets/s and both Nodes B and D transmit 1 packets/s. The probability that Nodes B and D transmit simultaneously is only 0.2. However, when Node C starts its traffic, Node C's transmission can increase the concurrency of Node B and Node D's transmission. This is because, when Node C finishes a transmission of its packet, both Nodes B and C start to compete for the channel and hence can start transmit almost simultaneously. Therefore, prediction of available bandwidth based on the free channel bandwidth or the delay measured before Node C starts its traffic is inaccurate since neither captures the increased amount of concurrent transmissions of Nodes B and C after Node C starts its traffic.

Although all the methods discussed in this paper are inaccurate since they are all based on the assumption that Node C's neighbors can hear each other, we believe that the degree of inaccuracy may be limited if Node C has more neighbors. Assume that Node C has N neighbors evenly distributed in its carrier-sensing range. Even for the nodes that are located at the edge of Node C, they can heard at least half of Node C's neighbors. Those nodes that are closer to Node C can hear a even larger number of Node C's neighbors. Therefore, the impact of Node C's false assumption is limited.

The analysis in Sections 7.1 and 7.2 shows that both our bandwidth allocation model and existing methods are inaccurate in multi-hop networks due to the different neighbor sets observed by nodes. However, since beyond the cases discussed in this section, the existing methods also have additional limitations (see Section 8), the degree of inaccuracy of their bandwidth predictions is much higher than our prediction model. Since the inaccuracy of our prediction model is limited, we expected that our model still produces accurate enough bandwidth predictions to have practical usage.

8. Limitations of bandwidth prediction approaches

The current approaches to available bandwidth prediction each have unique limitations. Our analysis shows that the delay and free bandwidth models are only able to estimate local achievable bandwidth for a flow. In addition, none of the three methods can capture the network under all network states, which is confirmed by our simulation results in Section 9.

8.1. Delay model

In [2,8,11,19,20], the available bandwidth at Node i is predicted using the average packet transmission delay at the MAC layer, Δ_i, which is the period of time between when a packet is ready to be transmitted at the MAC layer and the actual successful packet transmission. The available bandwidth of Node i is approximated as $\frac{1}{\Delta_i}$, with some additional adjustment for different packet sizes. However, the measurement does not consider how the transmissions of Node i may affect the delay or throughput of neighboring nodes. Hence, the measurement is only valid for estimating local achievable bandwidth. Neighborhood available bandwidth is not considered in this model.

In addition, this heuristic introduces inaccuracies for even predicting local achievable bandwidth. Although $1/\Delta_i$ represents the service rate of the network for the node, the following analysis shows that Δ_i is not independent of Node i's transmission rate. Hence, the estimation of local achievable bandwidth performed before Node i starts transmitting a flow does not reflect the actual local achievable bandwidth of the flow. When Node i increases its transmission rate, its Δ_i increases. Therefore, $\Delta_i < \Delta_{i,\text{sat}}$, where $\Delta_{i,\text{sat}}$ is the packet transmission delay when Node i is saturated. Hence, using Δ_i under low transmission rates to estimate the local achievable bandwidth, which is achieved when Node i is saturated, can be over-optimistic.

To show that Δ_i increases when Node i increases its transmission rate, note that Δ_i includes three parts: the time of Node i's backoff slots, the time consumed by transmissions from other nodes during Node i's backoff procedure and Node i's own transmission time. Therefore,

$$\Delta_i \approx B_i \times aSlotTime + B_i \sum_{j=1}^{n} \frac{\tau_j L_j}{C} + \frac{L_i}{C}, \tag{45}$$

where B_i is the average number of backoff slots before a transmission. First, B_i increases as Node i becomes saturated. When Node i is not saturated, after it finishes a transmission and the backoff procedure following the transmission, the next packet may still not have arrived. When the next packet arrives and Node i sees an idle channel, Node i does not need to perform a backoff procedure before it starts a new transmission. In this case, $B_i = 0$. However, when Node i becomes saturated, it must always backoff before it starts a transmission. Therefore, B_i increases as Node i becomes saturated. Second, the probability that some other nodes transmit during Node i's backoff procedure also increases as Node i becomes saturated. Assuming Node k is another non-saturated node in the network, from Eq. (8), the probability that Node k transmits in a randomly chosen virtual time slot becomes:

$$\begin{aligned} \tau_k &= \frac{P_k}{1-\phi_k} = \frac{R_k \sum_{j=1}^{n} P_j L_j}{(\sum_{j=1}^{n} S_j)(1-\phi_k)}, \\ &= \frac{R_k(\sum_{j=1, j\neq i}^{n} P_j L_j + P_i L_i)}{(\sum_{j=1}^{n} S_j)(1-\phi_k)}. \end{aligned} \tag{46}$$

As Node i becomes saturated, its P_i increases. Therefore, τ_k increases as shown in Eq. (46). Similarly, all other non-saturated nodes in the network also increase their τ as Node i increases its P_i. Therefore, $\sum_{j=1}^{n} \frac{\tau_j L_j}{C}$ in Eq. (45) increases, indicating more transmissions from other nodes during Node i's backoff procedure. Hence, $\Delta_{i,\mathrm{sat}} > \Delta_i$. Therefore, using packet delay to predict local achievable bandwidth is over-optimistic when there are many unsaturated nodes in the network.

8.2. Free bandwidth model

In [4], the available bandwidth at a Node i is approximated as the free channel bandwidth. However, this estimation is also only valid for local achievable bandwidth since it does not include the effects of Node i's flow on neighboring flows' throughput. To demonstrate this, we show a simple simulation using NS2 [9]. In this scenario, there are six mobile hosts positioned as in Fig. 11. The MAC layer protocol is IEEE 802.11 with 250 m radio transmission range and 550 m carrier-sensing range. The bandwidth of the wireless channel is 2 Mbps. Node C and Node E are each other's neighbors in carrier-sensing range. Node A is Node C's neighbor and is out of Node E's carrier-sensing range. Three 133 packets per second CBR flows with packet size of 512 Bytes are established between Nodes A and B, Nodes C and D and Nodes E and F. Due to the overhead of the MAC layer RTS-CTS-DATA-ACK handshake and collisions, each flow requires about 930Kbps channel bandwidth. At 1 s, Node A initiates Flow 1 to Node B. At 40 s, Node C initiates Flow 2 to Node D. Finally, at 80 s, Node E initiates Flow 3 to Node F. Fig. 12 shows the changes in free channel bandwidth at each source node as the three flows start successively. Fig. 13 shows the throughput and delay of each flow over time.

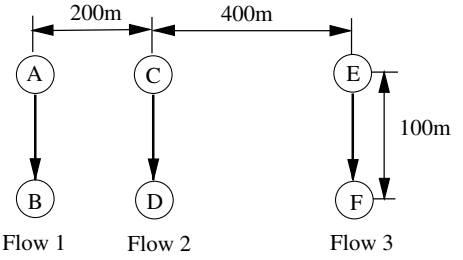


Fig. 11. Simulation topology.

As shown in Fig. 12, after Flows 2 starts, Node E has 1.07 Mbps bandwidth that is not consumed by the contention from Flow 2. Therefore, to Node E, there is 1.07 Mbps free channel bandwidth, which indicates enough local achievable bandwidth for Flow 3. Therefore, when Flow 3 starts, it can get its desired throughput and delay as shown in Fig. 13. However, since no consideration is given to neighborhood available bandwidth, the contention from Flow 3 actually decreases the throughput of Flow 2 by 20% and increases the delay of Flow 2 dramatically. Therefore, enough free channel bandwidth can only capture the local achievable bandwidth, but cannot capture the neighborhood available bandwidth.

In addition, even for local achievable bandwidth, the free bandwidth model cannot provide accurate estimation. This is due to the fact that free bandwidth does not equal the local achievable bandwidth

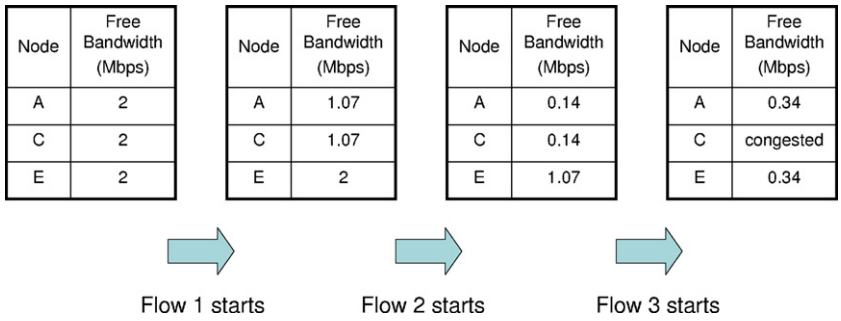


Fig. 12. Changes of free channel bandwidth.

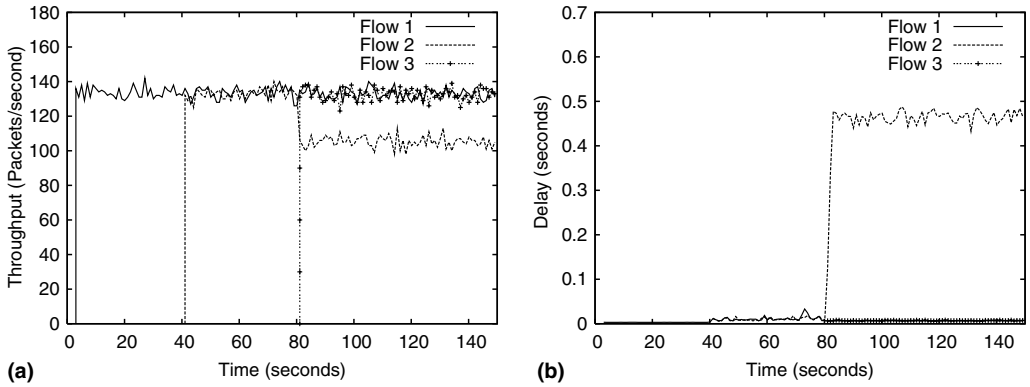


Fig. 13. Throughput and delay of Flows 1, 2 and 3.

to a flow in a contention-based IEEE 802.11 network, since newly arrived realtime flows should be allowed to "push down" the throughput of existing flows with lower priorities (e.g., best effort flows) through contention. However, if the free bandwidth model is used, a best effort flow can occupy all of the channel time and prevent the admission of any realtime traffic. Therefore, using free bandwidth to predict the local achievable bandwidth to a node can be over-pessimistic, especially in a heavily loaded network.

8.3. Saturation model

Similar to our bandwidth allocation model, the saturation model predicts available bandwidth by extending a bandwidth allocation model in a single-hop network to a multi-hop environment. Therefore, it can predict both local achievable bandwidth and neighborhood available bandwidth. However, the bandwidth allocation model in saturation model assumes that every nodes in the network is saturated [3,5,7,13,15]. This saturated network state is not necessarily true and for a network that provides throughput guarantees to flows, this saturated network state should be avoided. Therefore, this prediction method can be often over-pessimistic, especially in a lightly loaded network.

9. Evaluation

Our evaluation of MPARC includes four parts. First, since both the free bandwidth and delay models only estimate local achievable bandwidth, to be fair to these models, we compare the accuracy of MPARC's prediction of local achievable bandwidth with the other three methods for local achievable bandwidth estimations. Second, we examine the accuracy of MPARC's admission control and rate policing by comparing it with admission control using the other three models. Third, we demonstrate MPARC's ability to support admission control and rate policing in multi-priority networks. Finally, we examine the impact of mobility on the performance of MPARC.

NS2 is used for all simulations. The channel transmission rate is 2 Mbps, the transmission range is 250 m, and the carrier-sensing range is 550 m.

DSR [10] is used as the routing protocol. The implementations of MPARC, the delay model, the saturation model and the free bandwidth model are as follows. As discussed in Section 6.1, in MPARC, each node broadcasts its traffic information to its one-hop neighbors for every 2 s, which also carries the traffic information of its two-hop neighbors that it has learned from listening to other nodes broadcasts. By listening to broadcast of traffic information from its neighbors, every node can get the necessary information for performing admission control and rate policing based on the approach discussed in Section 5. Similar to MPARC, the saturation model also uses periodic message exchanges between neighbors to collect the number of active nodes in each node's carrier-sensing range so that prediction of priority-based available bandwidth can be made. In the delay model, to make priority-based estimation of available bandwidth, each node measures the average packet transmission delay for each priority classes by periodically sending a probing packet for each priority class to some randomly selected neighbors. In the free bandwidth model, each node monitors the amount of free channel time in its channel and uses it to predict available bandwidth.

9.1. Local achievable bandwidth estimation

To compare MPARC's local achievable bandwidth estimation with the free bandwidth, delay and saturation models under different MAC layer configurations, traffic is chosen from six priority classes with different minimum contention window sizes W. Table 1 shows the W configurations for the classes. The $CW^{\max}$ for all classes is 1023. In each simulation, a CBR flow y is placed at the center of the simulation area. Before flow y starts, the local achievable bandwidth of flow y is predicted. The four local achievable bandwidth prediction methods are used and compared to the actual maximum throughput of the flow to measure the accuracy of the prediction methods. In all simulations, flow y's competing flows are randomly placed around the network and their priorities are randomly chosen between 0 and 5. Fig. 14 shows an example random topology, where flow y has five hops and competes with 10 other flows. The number of competing flows ranges from 2 to the number of flows where using any more flows results in severe network congestion so that DSR is unable to find any route for flow y.

Table 1
W configurations of priority classes

Priority	5	4	3	2	1	0
W	7	15	31	63	127	255

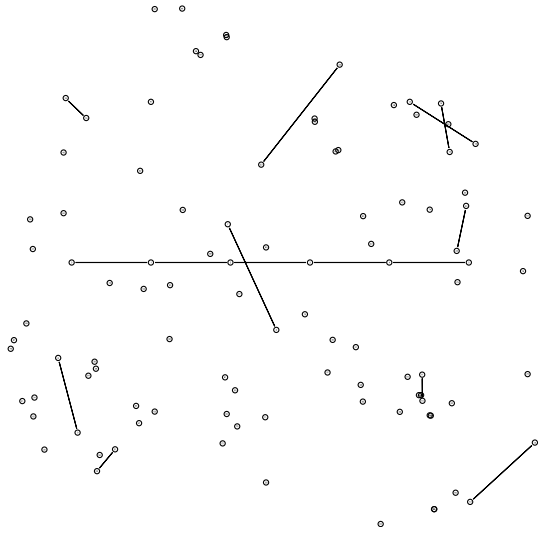


Fig. 14. Example of simulation topologies: In the center of the graph is the five-hop flow y (the horizontal line). The other 10 one-hop flows are the competing flows.

The first set of simulations are run in 550 m × 550 m networks. The number of competing flows with flow y ranges from 2 to 14. Each of the competing flows transmits ten 512 Byte CBR packets per second. Fig. 15 shows both the predicted and the measured achievable throughput. Predictions from our model are very close to the actual achievable throughput of flow y, while the delay model, the free bandwidth model and the saturation model all provide inaccurate predictions. As expected based on our discussions in Section 8, the saturation model's prediction is too conservative and its performance improves as the network load increases. The free bandwidth model's prediction is too conservative as well, especially when the network load is large. The delay model's prediction is too optimistic.

To examine the impact of multi-hop networks on the prediction accuracy of MPARC, in the second set of simulations, the network size is increased to 750 m × 750 m. Fig. 16 shows the predictions and the measured achievable throughput. Compared to Fig. 15, the prediction accuracy of our model is affected by the multi-hop networks due to the reasons discussed in Section 8. However, in most cases,

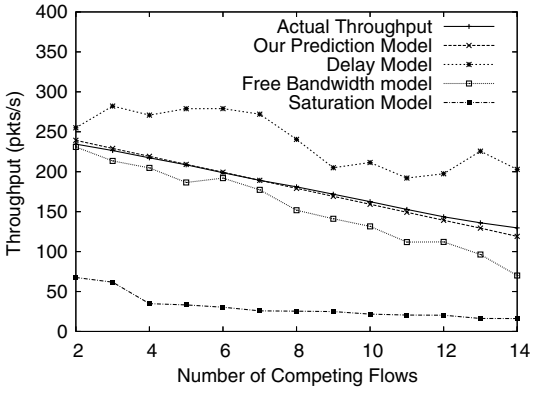


Fig. 15. 550 m × 550 m networks.

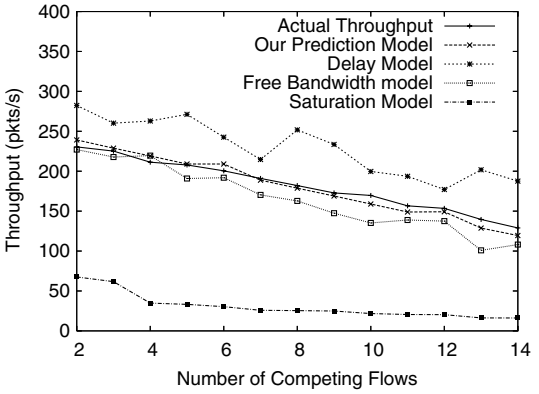


Fig. 16. 750 m × 750 m networks.

MPARC's performance is still better than the existing methods. To further confirm this, in the third set of simulations, we examine larger multi-hop networks and vary the hop count of flow y to measure the performance of our prediction model.

In the third set of simulations, the number of hops of flow y ranges from 1 hop to 5 hops. Only the results of the 1-hop simulations and 5-hop simulations are shown. The results for other hop counts are similar. Each simulation runs for 100 s in one of the 300 1000 m × 1000 m random networks with 100 nodes. The background traffic in these networks ranges from 1 to 16 randomly located active nodes, each sending CBR traffic with a rate uniformly distributed in [1, 50] 512 Byte packets/second.

To understand the average behavior of different prediction methods, two parameters are used to evaluate the predictions generated in n simulations: the standard deviation (SD) of the prediction error and the Mean (M) of the prediction error:

$$SD = \sqrt{\frac{\sum_{i=1}^{n}\left(\frac{Prediction_i - ActualThroughput_i}{ActualThroughput_i}\right)^2}{n-1}}, \quad (47)$$

$$M = \frac{1}{n}\sum_{i=1}^{n}\frac{Prediction_i - ActualThroughput_i}{ActualThroughput_i}. \quad (48)$$

SD measures the prediction accuracy, where a small SD indicates that the prediction is close to the actual achievable bandwidth. M shows the direction of the estimation error. A positive M indicates that the estimation tends to be larger than the actual local achievable bandwidth, while a negative M indicates that the estimation tends to be smaller.

Figs. 17 and 18 show the SD and M of flow y when it is a five hop flow. Figs. 19 and 20 show SD and M when flow y is a 1-hop flow. These figures show that our prediction model has the smallest standard deviation (SD) in most circumstances. In addition, these figures further confirm our analytical results from Section 8. In a lightly loaded network,

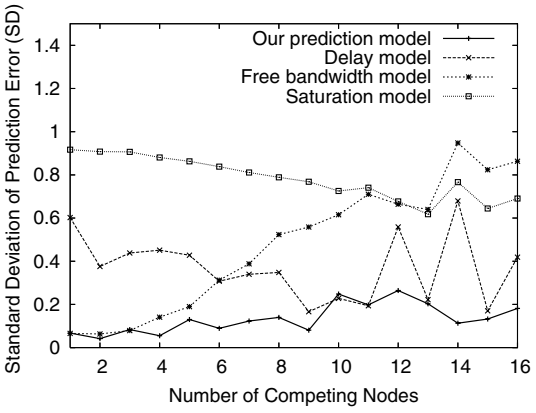


Fig. 17. SD for 5-hop flow.

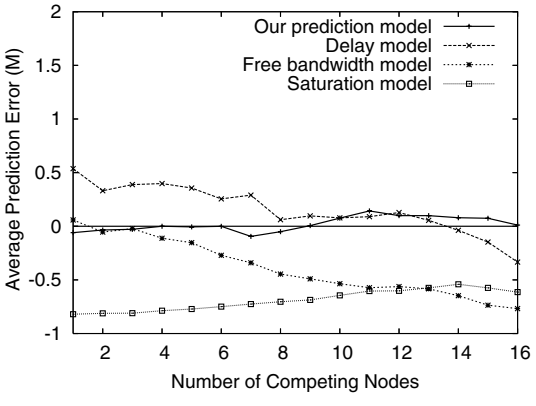


Fig. 18. M for 5-hop flow.

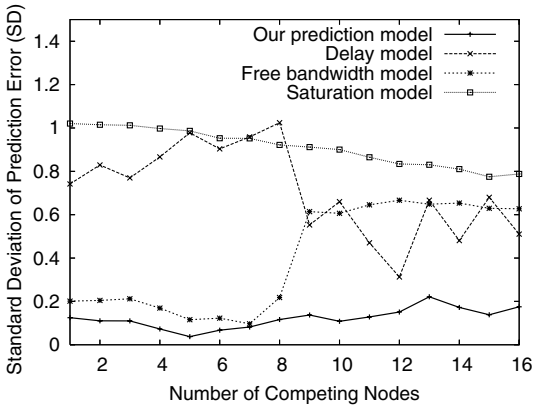


Fig. 19. SD for 1-hop flow.

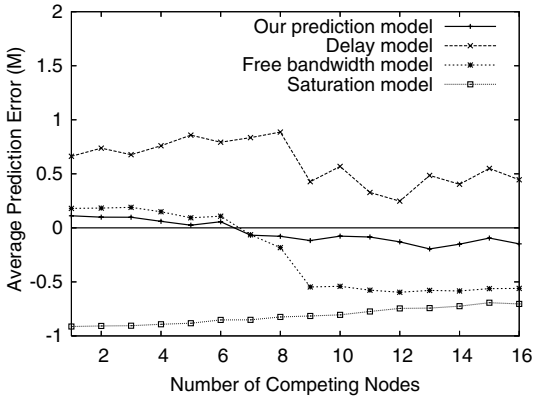


Fig. 20. *M* for 1-hop flow.

the saturation model is over-pessimistic, the delay model is over-optimistic and the free bandwidth model is accurate. As the load in the network increases, the free bandwidth model becomes over-pessimistic while the performance of both the saturated and delay models improves. When the network becomes heavily loaded, the delay and saturation models are better than the free bandwidth model since the network is close to saturated at this point. Overall, our model-based estimation method consistently provides better estimates than the other methods in lightly and heavily loaded networks.

9.2. Available bandwidth estimation and the accuracy of admission control and rate policing

In this section, we demonstrate that due to the accurate estimation of local achievable bandwidth and the inclusion of the concept of neighborhood available bandwidth, MPARC is able to make accurate estimations of available bandwidth. These accurate estimations enable MPARC to keep throughput guarantees to admitted flows, while maintaining high network utilization.

To demonstrates MPARC's ability to maintain the throughput of admitted realtime flows, five randomly generated topologies are used, each is $1000\text{ m} \times 1000\text{ m}$ square with 50 randomly positioned nodes. The simulations run 100 s. From the beginning to 55 s, for every 5 s, a new realtime CBR flow with 512 Byte packets and randomly selected rates between $[10, 50]$ packets per second performs admission control. The sources and destinations of all flows are randomly selected. Fig. 21 shows the violation of throughput guarantees, which is defined as the total throughput of all CBR flows minus the total desired rate of all CBR flows, in the first 55 s of the simulations. The delay model starts to show severe throughput violations at around 15 s due to its over-optimistic estimation of local achievable bandwidth and lack of neighborhood available bandwidth estimation. Although the free bandwidth model does not over-estimate local achievable bandwidth, it also starts to show large variations of throughput at 30 s due to its lack of estimations of neighborhood available bandwidth. Since both the saturation model and MPARC consider neighborhood available bandwidth, neither has throughput violations. However, Fig. 22 shows that, before 55 s of the simulations, the total network throughput under the saturation model is much smaller than the throughput under other methods. This is due to the saturation model's severe underestimation of both the local achievable bandwidth of flows and the neighborhood available bandwidth, which leads to false rejection of realtime

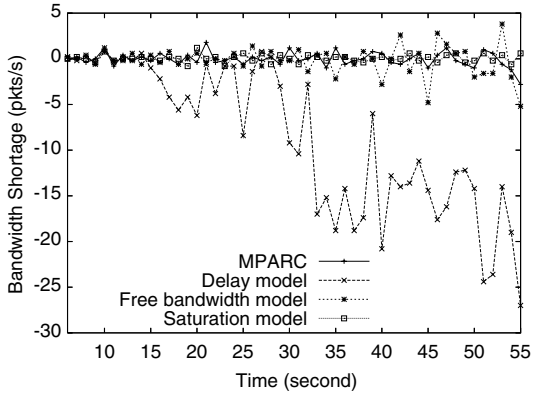


Fig. 21. Violation of throughput guarantee (no best effort flows).

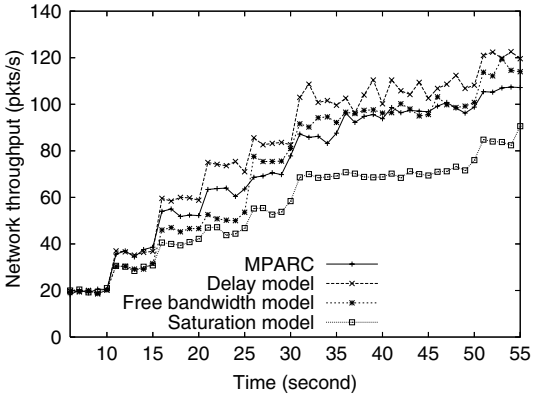


Fig. 22. Network utilization (no best effort flows).

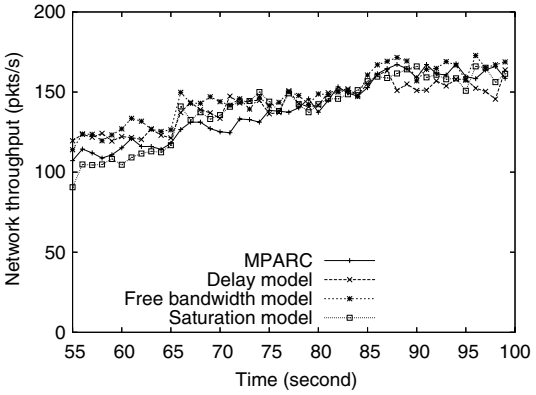


Fig. 24. Network utilization (with best effort flows).

flows. These unnecessary rejections reduce network utilization and limit the number of realtime flows that the network is able to carry. Overall, MPARC is the only protocol among the four methods that can use admission control to guarantee the throughput of realtime flows while maintaining high network utilization.

After 55 s, for every 5 s, a new best effort FTP flow starts. As shown in Fig. 23, the free model and delay model show severe violations because they does not have the rate policing mechanism which is based on the estimation of neighborhood available bandwidth. Hence, they do not regulate best effort traffic to protect the throughput of realtime flows. However, MPARC and the saturation model can effectively keep the throughput guarantees to realtime flows due to accurate rate policing of best effort flows. This accurate rate policing avoids over-penalizing of best effort traffic so that both MPARC and the saturation model can maintain network throughput that is comparable to the free model and the delay model as shown in Fig. 24. Since the saturation model may reject realtime flows unnecessarily as demonstrate in the first 55 s, MPARC is the only method that always achieves high network throughput while maintaining throughput guarantees to realtime traffic.

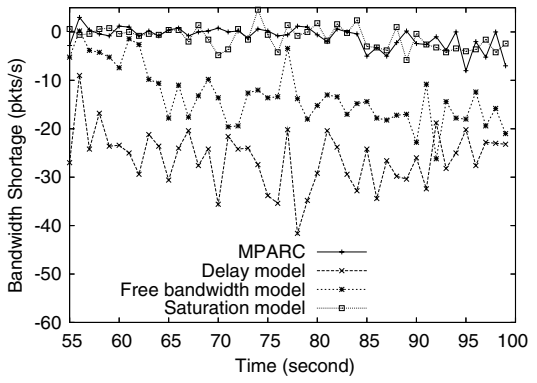


Fig. 23. Violation of throughput guarantee (with best effort flows).

9.3. Supporting multi-priority flows

When there are multiple priorities of flows, admission control decisions should be based on flow priorities. To investigate MPARC's ability to support admission control in such cases, we run two simulations. Each simulation has five realtime flows that transmit 200 512 Byte packets/s. The rate of the flows are deliberately set large enough so that no two flows can achieve their desired rates simultaneously.

In the first simulation, the five realtime flows start consecutively with increasing priorities. Fig. 25 shows the violation of throughput guarantees to each admitted flow. As a higher priority flow arrives, if this flow can achieve its desired bandwidth by competing with existing flows, MPARC and the saturation model admit the flow even if the new flow may degrade the throughput of existing lower priority flows. The throughput of the highest priority flow is always maintained in MPARC and the saturation model. The delay model, however, admits all newly arrived flows even if the new flow cannot achieve its desired rate, resulting in the violation of throughput guarantees to every flow. Since the free model does not recognize priority, it only admits the first flow and rejects

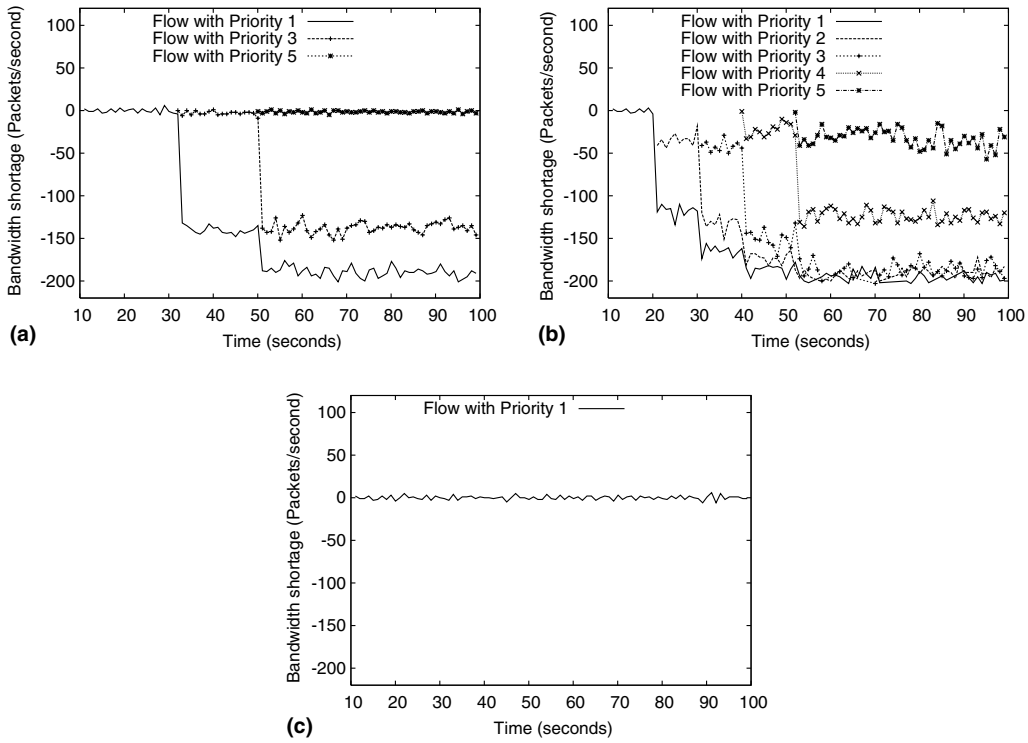


Fig. 25. Per flow violation of throughput guarantee (increasing priority flows). (a) MPARC and saturation model, (b) delay model and (c) free model.

all later flows even if the later flows have higher priorities.

In the second simulation, the five CBR flows start with decreasing order of priorities. Since the first admitted flow has the highest priority, the later lower priority flows should be rejected to protect the throughput of the first flow. Fig. 26 depicts the violation of throughput guarantees to admitted flows. Since the saturation model, free model and MPARC all only admit the first flow, they show no violation of throughput guarantees to the first admitted flow. The delay model, however, admits the first two flows and shows violation of throughput guarantees to both admitted flows. In conclusion, both MPARC and the saturation model can achieve priority-based admission control. However, as shown in Section 9.2, the saturation model may falsely reject realtime flows even if the network has

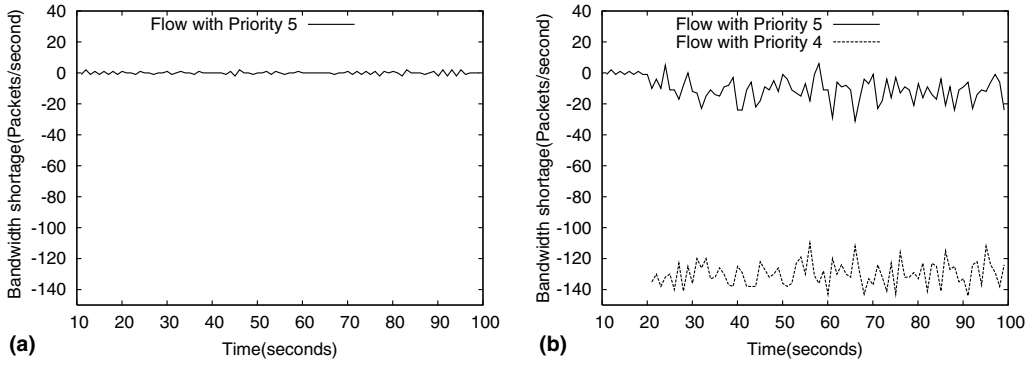


Fig. 26. Per flow violation of throughput guarantee (decreasing priority flows). (a) MPARC, saturation and free model and (b) delay model.

enough bandwidth. Hence, among all the four protocols, MPARC is the only protocol that can achieve accurate priority-based admission control and rate policing.

9.4. Impact of mobility

The simulations in the previous sections were all performed in static networks. However, when nodes are mobile, two flows may move into the interference ranges of each other and both flows' QoS may be degraded. Therefore, it is important to know if MPARC is able to provide QoS guarantees in mobile networks by rejecting realtime flows with low priorities and decreasing the transmission rates of best effort flows. We evaluate MPARC in eight scenarios. In each scenario, a random 1500 m × 300 m network is generated with 100 nodes in their initial positions. Then, the random waypoint model is used to create six different mobility patterns, where the maximum speed ranges from 0 to 25 m/s and the pause time is 10 s for all mobility patterns. For each scenario, 15 realtime flows with rates uniformly distributed between [1, 50] 512 Byte packets/second and 33 TCP flows are injected into the network during the 100-s simulation. Fig. 27 shows the aggregated throughput violations of all realtime flows in the network. MPARC provides very good throughput guarantees to realtime flows in all scenarios with the maximum throughput violation less than 0.1 packets/s. Although mobility seems to slightly increase the throughput violations in some scenarios, such as scenario 8, the increase is very small, which indicates that MPARC does not need to adjust its parameters, such as the traffic information broadcasting rate, based on the mobility of nodes.

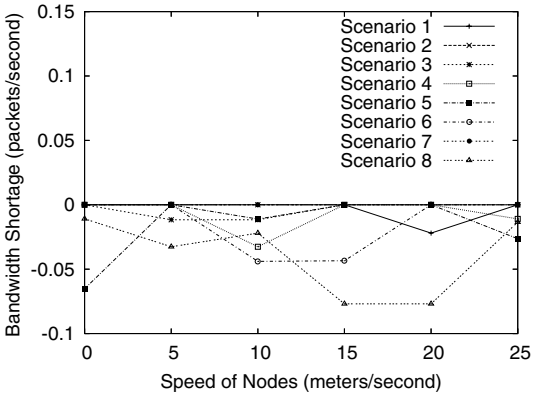


Fig. 27. Throughput violation of MPARC in mobile networks.

10. Conclusions and future work

In this paper, we use our novel bandwidth allocation model to design a joint admission control and rate policing protocol, MPARC. Since our bandwidth model captures bandwidth allocations in saturated, unsaturated or semi-saturated networks, our bandwidth model produces more accurate estimations of available bandwidth in the network. Through simulation, we show that using our available bandwidth estimation method, MPARC achieves accurate admission control of realtime traffic and rate policing of best effort traffic, which ensures that throughput guarantees for realtime flows are maintained and at the same time the network utilization is efficient. Since currently, our model only supports bandwidth allocations, questions related to delay, such as how to predict the packet delays of existing flows after the arrival of a new flow and the delay experienced by a new flow, are still open. Therefore, an important focus of our future work is to extend the bandwidth allocation model to express packet delays so that admission control decisions can be based on delay predictions.

Acknowledgments

This work was funded in part by a Vodafone-US Foundation Graduate Fellowship and National Science Foundation Grant 0081308. The author would also like to thank Al Harris, Cigdem Sengul, Jun Wang and William Yurcik for their help and support in making this work possible.

Appendix A. Uniqueness of η

The solution to S_i is unique since the solution of (η, k) that can satisfy both Eq. (24) and Inequality (25) is unique. This can be proved as follows. If there are two solutions (η_1, k_1) and (η_2, k_2), with $k_1 < k_2$, then the following inequality holds:

$$\eta(k_1) < \eta^*_{x_{k_1+1}}, \tag{A.1}$$

$$\eta(k_2) \geqslant \eta^*_{x_{k_2}}. \tag{A.2}$$

Combining Inequality (A.1) and (A.2) with Eqs. (23) and (24), we get

$$\frac{\sum_{i=1}^{k_1} \frac{L_{x_i}}{W_{x_i}}}{1 - \sum_{i=k_1+1}^{n} \frac{R_{x_i} L_{x_i}}{C}} < \frac{C}{R_{x_{k_1+1}} W_{x_{k_1+1}}}, \tag{A.3}$$

$$\frac{\sum_{i=1}^{k_2} \frac{L_{x_i}}{W_{x_i}}}{1 - \sum_{i=k_2+1}^{n} \frac{R_{x_i} L_{x_i}}{C}} \geqslant \frac{C}{R_{x_{k_2}} W_{x_{k_2}}}. \tag{A.4}$$

According to Inequality (A.3), we have

$$\sum_{i=1}^{k_1}\frac{L_{x_i}}{W_{x_i}} < \frac{1}{R_{x_{k_1+1}}W_{x_{k_1+1}}}\left(C-\sum_{i=k_1+1}^{n}R_{x_i}L_{x_i}\right) = \frac{1}{R_{x_{k_1+1}}W_{x_{k_1+1}}}\left(C-\sum_{i=k_2+1}^{n}R_{x_i}L_{x_i}-\sum_{i=k_1+1}^{k_2}R_{x_i}L_{x_i}\right). \tag{A.5}$$

According to Inequality (A.4), we have

$$\sum_{i=1}^{k_1}\frac{L_{x_i}}{W_{x_i}} \geqslant \frac{1}{R_{x_{k_2}}W_{x_{k_2}}}\left(C-\sum_{i=k_2+1}^{n}R_{x_i}L_{x_i}\right)-\sum_{i=k_1+1}^{k_2}\frac{L_{x_i}}{W_{x_i}} \tag{A.6}$$

$$= \frac{1}{R_{x_{k_2}}W_{x_{k_2}}}\left(C-\sum_{i=k_2+1}^{n}R_{x_i}L_{x_i}-\sum_{i=k_1+1}^{k_2}R_{x_i}L_{x_i}\frac{R_{x_{k_2}}W_{x_{k_2}}}{R_{x_i}W_{x_i}}\right) \tag{A.7}$$

Note that since $\eta^*_{x_i} < \eta^*_{x_j}$ if $i<j$, combining with Eq. (23), we have

$$\frac{1}{R_{x_i}W_{x_i}} < \frac{1}{R_{x_j}W_{x_j}} \quad \forall i<j. \tag{A.8}$$

Therefore, $\frac{R_{x_{k_2}}W_{x_{k_2}}}{R_{x_i}W_{x_i}} < 1\ \forall i<k_2$. Hence Inequality(A.7) can be further derived to be

$$\sum_{i=1}^{k_1}\frac{L_{x_i}}{W_{x_i}} \geqslant \frac{1}{R_{k_2}W_{k_2}}\left(C-\sum_{i=k_2+1}^{n}R_{x_i}L_{x_i}-\sum_{i=k_1+1}^{k_2}R_{x_i}L_{x_i}\frac{R_{x_{k_2}}W_{x_{k_2}}}{R_{x_i}W_{x_i}}\right) \tag{A.9}$$

$$\geqslant \frac{1}{R_{k_2}W_{k_2}}\left(C-\sum_{i=k_2+1}^{n}R_{x_i}L_{x_i}-\sum_{i=k_1+1}^{k_2}R_{x_i}L_{x_i}\right) \tag{A.10}$$

$$\geqslant \frac{1}{R_{k_1+1}W_{k_1+1}}\left(C-\sum_{i=k_2+1}^{n}R_{x_i}L_{x_i}-\sum_{i=k_1+1}^{k_2}R_{x_i}L_{x_i}\right) \tag{A.11}$$

Comparing Inequality (A.5) with (A.11), we get a paradox. Therefore, it is impossible that there are two solutions of (η,k). Hence, the solution of (η,k) is uniquely defined by Eq. (24) and Inequality (25).

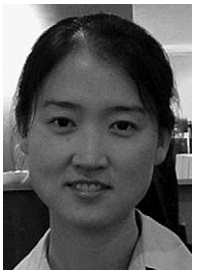
Appendix B. Notation

$\mathcal{N}$	set of transmitting nodes
n	number of competing nodes
CW	contention window size
$CW^{\min}$	minimum contention window size
W_i	minimum contention window size
$CW^{\max}$	maximum contention window size
S_i	bandwidth allocated to Node i
P_i	probability that Node i successfully transmits a packet in a virtual slot
Subscript sat	saturated node
Subscript $\overline{\text{sat}}$	unsaturated node
R_i	the packet arrival rate at Node i
L_i	frame size of Node i
τ_i	probability that Node i transmits in a randomly chosen virtual time slot
ϕ_i	collision probability of Node i's transmission
C	the maximum throughput of the channel
m	the number of collisions that are needed for the contention window size to reach $CW^{\max}$
N_1	saturated node set
N_2	unsaturated node set
ρ	the average number of bits transmitted in a virtual time slot
η	the part of ρ that does not depend on to τ_i or ϕ (see Eq. (11))
η^*_i	the threshold value of η at when node i is at the edge of turning from unsaturated to saturated (see Eq. (23))
Δ_i	average packet transmission delay at the MAC layer
$\Delta_{i,\text{sat}}$	the packet transmission delay when Node i is saturated
U^{l}_f	local achievable bandwidth of flow f at a node
U^{n}_f	neighborhood available bandwidth of flow f at a node
α	number of nodes along the route of flow f that are also in a node's carrier-sensing range
R^{l}_f	throughput of flow f when it just achieves its local achievable bandwidth
R^{n}_f	throughput of flow f when it just achieves its neighborhood available bandwidth

References

[1] Imad Aad, Claude Castelluccia, Differentiation mechanisms for IEEE 802.11, in: IEEE INFOCOM, 2001.
[2] Gahng-Seop Ahn, Andrew Campbell, Andras Veres, Li-Hsiang Sun. SWAN: service differentiation in stateless wireless ad hoc networks, in: IEEE INFOCOM, 2002.
[3] Albert Banchs, Xavier Perez-Costa, Daji Qiao, Providing throughput guarantees in IEEE 802.11e wireless LANs, in: Proceeding of the 18th International Teletraffic Congress (ITC-18), 2003.
[4] Michael G. Barry, Andrew T. Campbell, Andras Veres, Distributed control algorithms for service differentiation in wireless packet networks, in: IEEE INFOCOM, 2001.

[5] Giuseppe Bianchi, Performance analysis of the IEEE 802.11 distributed coordination function, IEEE Journal on Selected Areas in Communications 18 (3) (2000) 535–547.
[6] R. Braden, L. Zhang, S. Berson, S. Herzog, S. Jamin. Resource ReSerVation Protocol (RSVP)—Version 1 Functional Specification. RFC 2205, September 1997.
[7] Frederico Cali, Marco Conti, Enrico Gregori, Dynamic tuning of the IEEE 802.11 protocol to achieve a theoretical throughput limit, IEEE/ACM Transactions on Networking 8 (6) (2000) 785–799.
[8] Kai Chen, Klara Nahrstedt. Exact: an explicit rate-based flow control framework in manet. Technical Report UIUCDCS-R-2002-2286/UILU-ENG-2002-1730, 2003, p. 1.
[9] Kevin Fall, Kannan Varadhan, NS notes and documentation, in: The VINT Project, UC Berkely, LBL, USC/ISI, and Xerox PARC, 1997.
[10] David B. Johnson, David A. Maltz, Dynamic source routing in ad hoc wireless networksMobile Computing, vol. 353, Kluwer Academic Publishers, 1996.
[11] Manthos Kazantzidis, Mario Gerla, Sung-Ju Lee. Permissible throughput network feedback for adaptive multimedia in AODV MANETs, in: IEEE International Conference of Communications (ICC), 2001.
[12] Seoung-Bum Lee, Gahng-Seop Ahn, Xiaowei Zhang, Andrew Campbell, INSIGNIA: an IP-based quality of service framework for mobile ad hoc networks, Journal of Parallel and Distributed computing, Special issue on Wireless and Mobile Computing and Communications, 60 (2000) 374–406.
[13] Bo Li, Roberto Battiti, Performance analysis of an enhanced IEEE 802.11 distributed coordination function supporting service differentiation, in: International Workshop on Quality of Future Internet Service, 2003.
[14] Stefan Mangold, Sunghyun Choi, Peter May, Ole Klein, Guido Hiertz, Lothar Stibor, IEEE 802.11e wireless LAN for quality of service, in: Proceedings of European Wireless, 2002.
[15] P. Chatzimisios, V. Vitsas, A.C. Boucouvalas, Throughput and delay analysis of IEEE 802.11 protocol, in: Proceedings of the 5th IEEE International Workshop on Networked Appliances, 2002.
[16] Charles Perkins, Ad-hoc on-demand distance vector routing, in: MILCOM panel on Ad Hoc Networks, 1997.
[17] Charles Perkins, Pravin Bhagwat, Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers, in: ACM SIGCOMM, 1994.
[18] Dennis Pong, Tim Moors, Call admission control for IEEE 802.11 contention access mechanism, in: Proceedings of IEEE Globecom, 2003.
[19] Samarth H. Shah, Kai Chen, Klara Nahrstedt, Available bandwidth estimation in IEEE 802.11-based wireless networks, in: The 1st Bandwidth Estimation Workshop (BEst 2003), 2003.
[20] Samarth H. Shah, Kai Chen, Klara Nahrstedt, Dynamic bandwidth management for single-hop ad hoc wireless networks, in: Proceedings of IEEE International Conference on Pervasive Computing and Communications (PerCom), 2003.
[21] IEEE Computer Society, 802.11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.
[22] Jun Zhao, Zihua Guo, Qian Zhang, Wenwu Zhu. Performance study of MAC for service differentiation in IEEE 802.11 WLAN, in: Procceedings of IEEE Globecom, 2002.



Yaling Yang got her B.S. degree in Telecommunications in University of Electronic Science and Technology of China in 1999. She is currently a Ph.D. candidate of Computer Science in University of Illinois at Urbana-Champaign. Her research interests include resource management and QoS in wireless networks and network routing and congestion control.

Robin Kravets is currently an assistant professor at the Computer Science Department at the University of Illinois, Urbana-Champaign. She received her Ph.D. from the College of Computing, Georgia Institute of Technology in 1999. She is the head of the Mobius group at UIUC, which researches communication issues in mobile and ad hoc networking, including power management, connectivity management, transport protocols, admission control, location management, routing and security. Her research has been funded by various sources, including the National Science Foundation and HP Labs. She actively participates in the mobile networking and computing community, both through organizing conferences and being on technical program committees. She is currently a member of the Editorial Board for IEEE Transactions on Mobile Computing and Elsevier Ad Hoc Networks Journal and was an Associate Editor of MC2R: Mobile Computing and Communications Review, a publication of ACM SIGMOBILE. She is also a member of the Steering Committee for WMCSA, the IEEE Workshop on Mobile Computing Systems and Applications.