Determining Intra-Flow Contention along Multihop Paths in Wireless Networks

Kimaya Sanzgiri Ian D. Chakeres[‡] Elizabeth M. Belding-Royer
Dept. of Computer Science
[‡]Dept. of Electrical & Computer Engineering
University of California, Santa Barbara
kimaya@cs.ucsb.edu, idc@engineering.ucsb.edu, ebelding@cs.ucsb.edu

Abstract

Admission control of flows is essential for providing quality of service in multihop wireless networks. In order to make an admission decision for a new flow, the expected bandwidth consumption of the flow must be correctly determined. Due to the shared nature of the wireless medium, nodes along a multihop path contend among themselves for access to the medium. This leads to intra-flow contention; contention between packets of the same flow being forwarded at different hops along a multihop path, causing the actual bandwidth consumption of the flow to become a multiple of its single hop bandwidth requirement. Determining the amount of intra-flow contention is non-trivial since interfering nodes may not be able to communicate directly if they are outside each other's transmission range. In this paper we propose two methods to determine the extent of intra-flow contention along multihop paths. The highlight of the proposed solutions is that carrier-sensing data is used to deduce information about carrier-sensing neighbors, and no high power transmissions are necessary. Analytical and simulation results show that our methods estimate intra-flow contention with low error, while significantly reducing overhead, energy consumption and latency as compared to previous approaches.

1. Introduction

Several applications that require either resource guarantees or priority network service have been envisioned for wireless multihop networks. To enable such service, the network needs to ensure that sufficient resources are available for a new flow before it is admitted. In other words, traffic admission into the network needs to be controlled.

To make an admission control decision, the network must first accurately estimate the resources that a flow will consume if admitted. Since bandwidth is an important resource for several applications, in this paper we focus on estimation of bandwidth consumption of a flow. This problem is complicated due to the shared nature of the wireless medium. A wireless node's transmissions consume bandwidth shared with other nodes in its vicinity since these nodes cannot simultaneously access the shared medium. More specifically, wireless transmissions consume bandwidth at all nodes within the carrier-sensing distance of the transmitting node. This carrier-sensing range is explained in detail in Section 2.1. Further, multiple nodes along a multihop path may be located within carrier-sensing distance of each other. This causes nodes to contend for medium access and prevents simultaneous transmissions. This in turn leads to intra-flow contention, i.e., contention between packets belonging to a single flow that are forwarded at different hops along a multihop path.

To calculate the intra-flow contention of nodes along the path, it is important to know the *contention count* of each node. The contention count (CC) at a node is the number of nodes on the multihop path that are located within carrier-sensing range of the given node [11]. Intra-flow contention has a significant impact on the bandwidth consumed by the flow. The effective bandwidth consumed by a flow at each node is the CC times the single hop flow bandwidth requested by the application. Hence, determination of the CC is an important part of making a correct admission control decision. (Note that other components, such as bandwidth availability determination, are also required for admission control. We only focus on CC calculation in this paper.)

Calculating the CC is difficult since there is no simple way for a node to determine its carrier-sensing neighbors, i.e. the set of nodes that are located within carrier-sensing range. One way may be to use high power transmissions, but these reduce spatial reuse and are expensive in terms of energy. In this paper, we propose two new approaches to determine the CC without using high power transmissions. Through simulation-based evaluation we show that both proposed methods outperform previous solutions in terms of overhead, power and delay, while computing the CC with low error.

The remainder of this paper is organized as follows. Section 2 provides background information on wireless transmissions, intra-flow contention and previous work. In Sec-

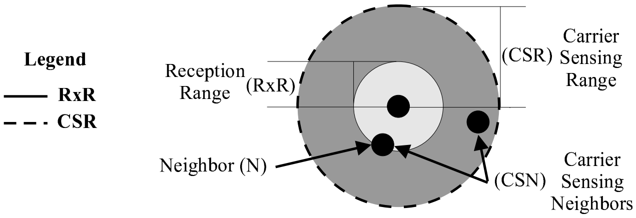


Figure 1. Notable ranges of IEEE 802.11 wireless communication.

tion 3 we describe two new approaches to determine the CC. In Sections 4 and 5 we evaluate the performance of the different techniques for CC calculation. Finally, Section 6 concludes the paper.

2. Background

In this section we discuss background information that is necessary to understand and analyze the solutions presented in this paper. Section 2.1 describes the notable distances for wireless communication. Section 2.2 explains the concept of intra-flow contention in detail, and Section 2.3 reviews previously proposed approaches for determining the CC.

2.1. Impacted Area

The maximum separation between a sender and receiver for successful packet reception is called the reception range (RxR), as shown in Figure 1[1]. Nodes within RxR of a particular sender can directly communicate with the sender and are considered its neighbors (N).

The maximum distance that a node can detect an ongoing packet transmission (carrier signal) is called the carrier-sensing range (CSR). This range is typically much larger than the reception range. Nodes that are within the CSR of a sender are called its carrier-sensing neighbors (CSN). These nodes detect a transmission but may not be able to decode the packet if they are outside RxR. In wireless MAC protocols based on CSMA, such as IEEE 802.11, all CSN of the sender are unable to initiate a packet transmission while the sender is transmitting because they sense the channel is busy. This helps avoid collisions at receivers.

2.2. Calculating the Contention Count

The contention count at a node is defined as the intersection of the set of nodes that lie on the multihop path with the set of carrier-sensing neighbors (CSN). Therefore, to calculate the CC at a particular node, the list of CSN and the identity of the nodes on the path (NoP) must be known. Given these, the CC for node i is:

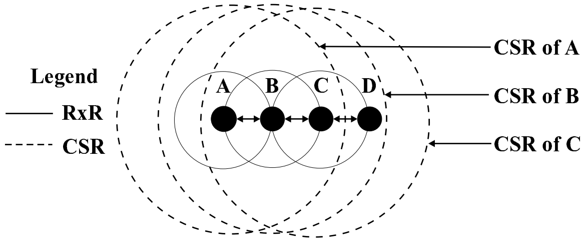


Figure 2. Example of intra-flow contention.

$$CC_i = |CSN_i \cap NoP| + 1 \tag{1}$$

The first term is the number of competing CSN, and one is added to account for the impact of node i itself. As an example, node A in Figure 2 communicates with node D via nodes B and C. Each packet transmitted by node A needs to be forwarded by nodes B and C in order to reach node D. However, nodes A, B, and C lie within carrier-sensing range of each other, and so only one of these nodes can transmit at any given time. The CC at each node in this example is therefore three. Note that the value for the CC may be different at each node along a multihop path, since it depends on the topology of the route.

Calculation of the CC is difficult, primarily because a node has no direct method for communication with other nodes that are farther away than the transmission range but within carrier-sensing range. Consequently, there is no straightforward method to determine the set of carrier-sensing neighbors. One solution is to use high power transmissions [11], but this method has several drawbacks, such as consumption of additional energy and an increase in collisions. In this paper, we propose two new mechanisms to detect the CSN that lie on a multihop path without the use of high power transmissions.

2.3. Related Work

QoS in wireless multihop networks is a popular area of research, and several QoS routing and admission control solutions have been proposed [1, 5, 7]. Many of these solutions completely ignore the effect of intra-flow contention. To the best of our knowledge, two approaches to determine contention count have previously been developed. In the following, each of these algorithms is described and their drawbacks are mentioned.

Ad hoc QoS On-demand Routing (AQOR): In AQOR [10], the authors ignore contention between multiple nodes that are located within carrier-sensing range. It is only considered that a single wireless node cannot transmit and receive messages simultaneously. As a result,

[1] Note: While we represent the transmission and carrier-sensing ranges as circles in this paper, in reality they are not perfect circles. Wireless signal propagation is influenced by many factors, including multipath interference, obstacles, and other phenomena.

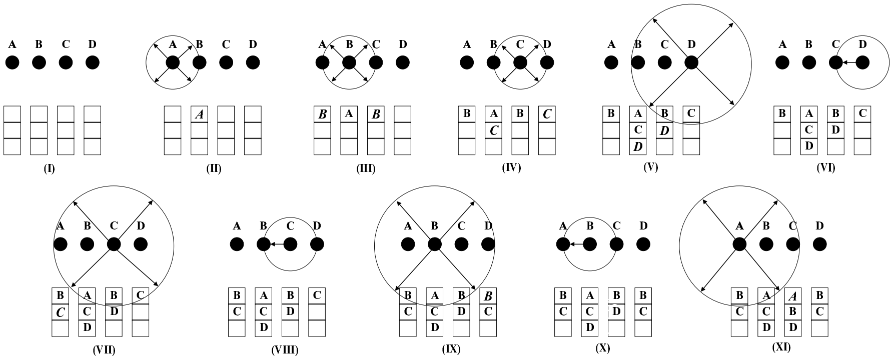


Figure 3. Contention count calculation with CACP.

given the per-hop flow bandwidth (BW_f) and assuming a bi-directional traffic flow, the flow bandwidth requirement at each node on the path is $2 * BW_f$, since each node must both receive and send packets for each flow. This simple formula does not work in the general case when the carrier-sensing range is larger than the reception range. For example, in Figure 2, the CC at each node is three, but AQOR computes it to be two.

Contention-Aware Admission Control Protocol (CACP): CACP [11] is an admission control solution for wireless multihop networks that takes intra-flow contention into consideration. Since data sessions are typically preceded by route discovery when using reactive routing protocols, admission control is integrated with the route discovery mechanism. In this paper, we focus only on CACP's mechanism for calculating the contention count.

In CACP, nodes use high power transmissions to communicate directly with their carrier-sensing neighbors. The example in Figure 3 illustrates how CACP determines the CC. In the figure, a list of the known CSN of each node is shown below the node during each step of the protocol operation. Node A, the source, needs a route to node D, the destination. Node A broadcasts a RREQ. Nodes B and C re-broadcast the request, appending their node ID prior to transmission. When node D receives the request, it starts the reply phase. At this point all nodes know their direct neighbors through the broadcast RREQ messages, but do not yet know their CSN.

In the reply phase, the destination sends a High Power Broadcast Message (HPBM) at a power high enough that all CSN can successfully receive it. The HPBM is received by all CSN of node D and contains the NoP list (the list of nodes the RREQ traversed, which was accumulated in the packet). Upon reception of the HPBM, nodes calculate their CC using their known CSN and the NoP list. However, the CC calculation may not be correct at this time, since all nodes do not yet know their CSN. For example, node C calculates the CC using Equation (1). In this case:

$$CC_C = |\{B, D\} \cap \{A, B, C, D\}| + 1 = |\{B, D\}| + 1 = 3 \qquad (2)$$

After sending the HPBM, the destination transmits the RREP message to the next hop toward the source. This transmission occurs at the regular power level. Each intermediate node, as well as the source, repeats this procedure. Each time a HPBM is received, nodes on the path have the opportunity to learn of a new CSN and recalculate their CC. Once the source sends the HPBM, all nodes know their CSN and are able to calculate the correct CC. For example on reception of the HPBM from node A, node C calculates its CC using Equation (1):

$$CC_C = |\{A, B, D\} \cap \{A, B, C, D\}| + 1 = 4 \qquad (3)$$

Although CACP's approach calculates the CC correctly, it has several drawbacks. First, CACP requires the use of high power messages at every node along a path to communicate with their CSN. High power transmissions require a capable radio. They are also very expensive in terms of energy since transmission power increases hyperbolically with increasing distance. This is a major drawback in wireless networks, where most devices are battery-powered and energy is a scarce resource. Second, high power transmissions impact a large area of the network. This reduces the spatial reuse of the medium and may increase collisions. Third, nodes on the path need to recalculate their CC each time a HPBM is received from another node on the path. This is inefficient. Fourth, nodes do not know the correct CC when they process the RREP message, as illustrated in the example earlier. As a result, a node cannot make an admission control decision when the reply is processed. The correct CC is known only when the HPBM is received from each CSN on the path. The admission control decision is therefore delayed until that time. This additional delay depends on the topology of the path, and in the worst case is proportional to the length of the path. Fifth, the RREP message needs to be delayed at each intermediate node in order to ensure that the node's HPBM is sent before the RREP. Since HPBMs are broadcast messages, their transmission is delayed at each node by a small random time (jitter) in order to reduce collisions. If the RREP is not delayed correspondingly, it may reach the source node before all the HPBMs have been sent and received, and the source may incorrectly

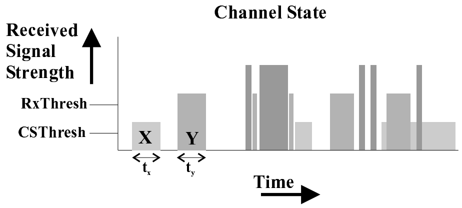


Figure 4. Diagram of received signal strength versus time.

admit the flow. Finally, CACP requires node IDs to be accumulated on routing packets, which increases the packet size and the routing load on the network.

In the following section, we describe two new approaches for determining the CC. Our proposed schemes do not require high power transmissions and address many of CACP's other drawbacks.

3. Proposed Solutions for Determination of the Contention Count

In this section, we propose two new approaches for determining the intra-flow contention count. The fundamental idea behind our approaches is to use carrier-sensing information from regular-powered transmissions to infer the CSN of each node. We first describe how carrier-sensing is performed and how we can use it to infer the needed information. Then we describe our two approaches. Like CACP, our proposed approaches are integrated with the route discovery procedure of reactive routing protocols. Note that the basic idea can also be applied to proactive routing environments with appropriate modifications.

3.1. Carrier Sensing and Packet Size Measurement

When a node transmits a packet, all nodes within carrier-sensing range can detect its carrier signal. The ability of a packet to be received depends upon its received signal strength, which varies at each receiver and is affected by the distance from the sender and other factors.

Figure 4 is a graph of received signal strength over time at a given node. If there are no ongoing transmissions and the channel is idle, the received power is small. When a transmission occurs at a node within carrier-sensing range, the received signal strength is greater than the carrier-sensing threshold (CS_{thresh}) and the receiving node is able to detect the packet. In the figure, the received signal strength of packet X is above CS_{thresh}, so the node can detect this packet transmission. If the strength of the received signal is greater than the reception threshold (Rx_{thresh}), the contents of the packet can be decoded; this happens when the receiver is within reception range of the sender. Referring to the figure, packet Y can be received and decoded by the node since its received signal strength exceeds Rx_{thresh}.

Given the signal strength measurements, a node can construct a graph showing the channel state over time, similar to the one in Figure 4. From this graph, it can determine the length of packets. For example, in Figure 4, the node measures the duration of the received signal corresponding to packet X. From this duration t_x, it can infer the length of packet X. Note that although packet X cannot be decoded its length can still be determined.

When simultaneous transmissions occur, the received signal strength of the packets overlaps. However the signal strength of the highest power packet dominates this measurement. The ability to correctly receive a packet in the presence of noise or other transmissions depends on the capture threshold of the wireless hardware. The capture threshold (C_{thresh}) defines the required proportion of signal power for two different signals such that the radio can properly receive the higher power signal [9]. For example, suppose the received signal strengths of two packets are P_x and P_y. A node can capture packet X if $P_x/P_y > C_{thresh}$. Similarly, if $P_y/P_x > C_{thresh}$, packet Y can be captured. If neither condition is true, neither packet X nor Y are receivable or discernible.

3.2. Pre-Reply Probe

In our first approach, called Pre-Reply Probe (PRP), nodes continuously monitor the received signal strength and record the durations of detected packets as described in the previous section. The packet duration information is soft-state, i.e. it times out after some interval and is deleted. The operation of PRP can be described through the example illustrated in Figure 5. In the figure, the table below each node contains the time durations of packets sensed by the node. Packet duration measurements that are not important to the CC calculation are not shown. Initially node A, the source, wants to find a route to node D, the destination. Node A generates a RREQ. The request is rebroadcast by each intermediate node as in the regular route discovery procedure. No additional processing is required during this phase.

When the destination receives the RREQ, it generates a Pre-Reply Probe Message (PRPM). The size of the PRPM message is randomly selected by the destination. This size in turn identifies a unique transmission duration, assuming all nodes use a common data rate. For example, consider that the random size selected by the destination results in a transmission duration t_y. The destination sends the PRPM to the next hop towards the source. This transmission is sensed by all nodes within carrier-sensing range of the destination. These nodes then add the value t_y to their carrier-sensing tables. Referring to Figure 5, the PRPM transmission by node D is sensed by nodes B and C, and both nodes record the duration t_y in their tables.

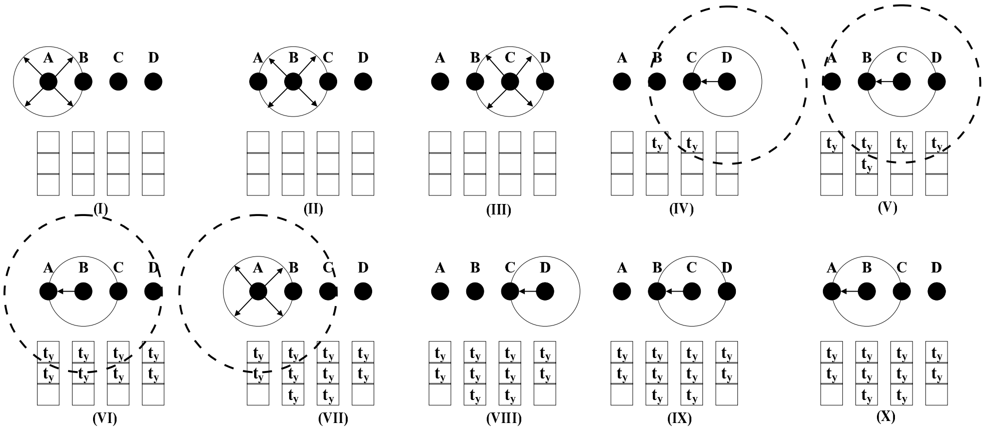


Figure 5. Contention count calculation with PRP.

Upon reception of a PRPM, intermediate nodes process the message by forwarding it to the next hop toward the source. The time duration of each transmission of the PRPM is recorded by all nodes located within carrier-sensing range of the sender. In Figure 5, transmission of the PRPM by node C is sensed and recorded by all other nodes, since they all lie within carrier-sensing range of C.

When the source receives the PRPM, it locally broadcasts the message one final time. This broadcast is required to indicate to all nodes along the path whether they are in the source's carrier-sensing range. The duration is sensed and recorded by the source's CSN. Once this final transmission occurs, each node has measured the duration of the PRPM messages of all its CSN that lie on the path.

After sending the PRPM, the destination waits for a small time interval and then sends a RREP to the source, as in the regular route discovery procedure. The RREP includes the size of the PRPM message that was transmitted previously. Upon receiving the RREP, each node on the path uses the PRPM size to calculate its CC by examining the duration of packets that were previously recorded. For each packet detected that matches the PRPM size from the RREP, the CC is increased by one. For example, node C knows that it heard a packet of size t_y transmitted three times, and it also transmitted the PRPM once. From this information, it determines its CC to be four. Each node along the path can accurately compute its CC in this manner.

The PRP approach alleviates many of the drawbacks of CACP. First, it determines the CC without high power transmissions. This results in energy-savings, better spatial reuse and reduced probability of collisions. Second, only the destination node introduces a delay in the forwarding of the RREP, unlike the per-hop delay introduced by CACP. This reduces the latency of determining the CC, as well as the route acquisition latency. Third, each node on the path knows the correct CC when the Route Reply is received and can immediately make an admission control decision. Moreover, each node needs to calculate the CC only once. Finally, the method adds only one additional field to the RREP, and does not require accumulation of node IDs on the route discovery messages.

The PRP method still has a few drawbacks. It requires an additional message (PRPM) to be transmitted during route discovery. This increases the network overhead. Also, the RREP is delayed at the destination node. This increases the route acquisition latency, and also delays admission control decisions. Finally, counting sensed packets of a particular duration can produce erroneous results in the case of retransmissions or collisions at the MAC layer. We address many of these concerns in our second approach.

3.3. Route Request Tail

Our second approach, Route Request Tail (RRT), removes the additional messaging and delay from the PRP approach. As in the previous approach, nodes record the sensed packet durations. However, instead of introducing a new packet, a *tail* is attached to RREQ packets in the RRT approach. This tail has a unique size at each node. In other words, at each node, the length of a RREQ packet is increased by an amount unique to that particular node. This increase in packet size serves to uniquely identify the RREQ transmission. The tail size can be randomly selected by each node. Alternatively, it could be derived from the node ID.

To describe the details of the RRT approach we provide the following example. In Figure 6, the table below each node lists the length of packets it has detected (packet length is inferred from the transmission duration). During route discovery the source, node A, creates a RREQ. It appends a tail of unique length to the packet. In addition to the tail, a field is inserted into the RREQ message. This field contains the size, S_a, of the packet including the tail. The source then broadcasts the RREQ. In the example, nodes B and C, which lie within node A's carrier-sensing range, record the size of the RREQ packet.

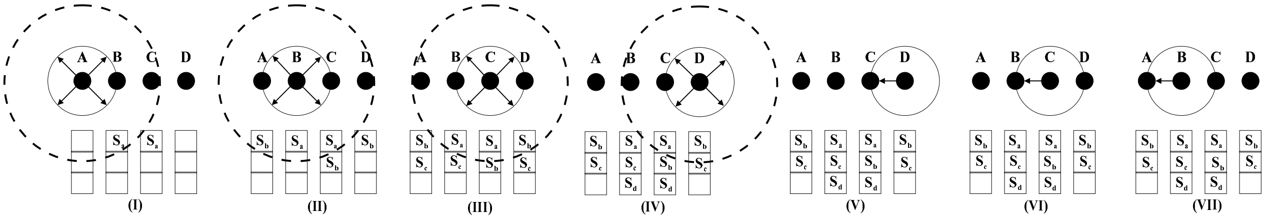


Figure 6. Contention count calculation with RRT.

Upon receiving the RREQ, each intermediate node removes the tail added by the previous node and attaches a new tail of a different size to identify itself. It also records the new packet size in the RREQ by appending it to the sizes recorded by previous nodes on the path. Thus, a list of random packet sizes is accumulated in the RREQ packet. In Figure 6, node B rebroadcasts the RREQ after replacing the tail, such that the new size of the packet is S_b. Each of node B's CSN (nodes A, C and D) records the packet of length S_b in its carrier-sensing table.

When the destination receives the RREQ, it repeats the procedure followed by the intermediate nodes and rebroadcasts the RREQ one more time. This is required to indicate to other nodes on the path whether they are within the destination's carrier-sensing range. Note that this is not necessary if the flow is uni-directional, i.e., if the destination is only going to receive data packets. Next, the destination generates a RREP. In the RREP, it includes the accumulated list of RREQ packet sizes, including that of itself. In the example, the destination, node D, places the list of packet sizes (S_a, S_b, S_c, S_d) in the RREP message. Node D then unicasts the RREP to node C.

When an intermediate node receives a RREP, it calculates its CC for the flow by examining its carrier-sensing table and looking for packets of sizes that match those indicated in the RREP. For each packet that matches a packet size from the RREP, the CC of the node increases by one. For example, in Figure 6, node B has seen packets of size S_a, S_b, S_c and S_d. Therefore, its CC is four.

The RRT approach retains most of the benefits of the PRP approach. It removes the extra messages needed by the PRP approach and instead increases the size of the RREQ. Transmission of a few extra bytes is less expensive than transmission of additional packets. The RREQ packet accumulates the various packet sizes generated by nodes on the path. Each packet size can be represented in one or two bytes. This is less expensive than accumulating node IDs (4 bytes). Finally, the RREP is not delayed. This results in quick route acquisition and admission decision propagation.

The drawback of the RRT approach is that larger packets have longer transmission times, and are hence more likely to suffer from collisions when the medium is heavily loaded. Collisions affect the packet duration measurements made by carrier-sensing neighbors, as explained in Section 3.1. This impacts the accuracy of the CC calculation.

4. Analytical Comparison

In this section, we present a simple analytical comparison of the three protocols: CACP, PRP and RRT. We compare the three protocols on the number and size of control packets transmitted, the number of CC calculations performed and the additional delay incurred in route discovery. Note that this performance is for a single route discovery. Table 1 presents the comparison.

As seen in Table 1, all three protocols transmit the same number of RREQ packets; this is equal to the number of nodes in the network (N) in most cases. The number of RREP packets is also the same for the three approaches, and is equal to length of the path (M). Additionally, CACP transmits M extra high power packets (HPBMs), one at each node along the selected path, while PRP requires M extra transmissions at regular transmit power (PRPMs). RRT does not transmit any extra messages.

In CACP, the RREQ packets accumulate the IDs of the nodes they traverse, so the size of the packets increases by M*I, where I is the size of the node ID. Similarly, RREQ packets in RRT accumulate the tail lengths. This causes the packet size to increase by M*J, where J is the size of a short integer (J < I). Additionally, the RREQ packet carries the tail appended by the last node traversed, which causes a further increase of T in the packet size. There is no increase in the size of RREQs in PRP. A corresponding increase in the RREP size occurs for CACP and RRT. The RREP in PRP must contain the length of the probe that was sent by the destination, hence the size increases by J.

Next, we look at the size of additional control messages. CACP HPBMs contain the list of node IDs on the path, and hence their size is M*I. PRPMs have a random size of S. RRT has no additional control messages.

The extra delay incurred in route acquisition is proportional to the length of the path in CACP, since the forwarding of the RREP is delayed at each intermediate node by a constant time (D1). In PRP, a constant extra delay (D2) occurs since the RREP is delayed by this value only at the destination node. RRT requires no additional delay. Finally, in CACP, each node calculates the contention count K times, where K is the final contention count value, since the CC calculation must be repeated each time an HPBM is received from a CSN on the path. In both PRP and RRT, the CC is calculated just once when the RREP is processed.

Approach	RREQ sent	RREP sent	Other control packet sent	RREQ size	RREP size	Other control packet size	Delay	CC calcs
CACP	N	M	M (High power)	Q + M*I	P + M*I	M*I	M*D1	K
PRP	N	M	M	Q	P + J	S	D2	1
RRT	N	M	0	Q + M*J + T	P + M*J	0	0	1

Variables:

N = *Number of nodes in the network*
M = *Number of nodes on the path*
Q = *Size of RREQ message*
P = *Size of RREP message*
I = *Size of nodeID*
J = *Size of integer*
S = *Size of PRPM (random)*
T = *Size of RREQ tail in RRT (derived from nodeid)*
$D1$ = *Delay at each hop between forwarding of HPBM and RREP in CACP*
$D2$ = *Delay at the destination between sending of PRPM and RREP in PRP*
K = *Contention count, i.e. the number of nodes on the path that are CSN*

Table 1. Contention count calculation overhead.

5. Simulation-based Evaluation

We compare the accuracy and overhead of the three protocols using simulation. The NS-2 simulator [3] is used for this purpose. We implement the three mechanisms by making appropriate extensions to the AODV-UU [6] NS-2 implementation of the AODV [8] routing protocol. AODV path accumulation [4], where the IDs of intermediate nodes are accumulated on AODV routing packets, is used to enable CACP to discover the identities of all nodes on a path.

In addition to the three protocols, we also implement a fourth mechanism that calculates the contention count from a global view of the network. This method, which we call the *Ideal* method, always computes the CC accurately through global knowledge and provides us with a reference for determining the accuracy of the other protocols. We note that such a method cannot be implemented in a real network because of the impracticality of global knowledge.

In the following sections, we describe our simulation parameters and define the performance metrics used to compare the protocols. This is followed by a description of the simulation scenarios and the performance results obtained.

5.1. Simulation Parameters

We use the two ray ground propagation model and IEEE 802.11 as the MAC protocol. The reception range is set to 250m and the capture threshold is 10. To prevent collisions of received packets, the carrier-sensing range should be set to (RxR + RID) [2], where RxR is the reception range and RID (receiver interference distance) is the minimum separation between a receiver and another sender, such that the sender's transmissions do not affect the receiver's ability to receive packets from its own sender. With our settings for reception threshold, capture threshold and regular transmission power, RID turns out to be 440m. We therefore set the carrier-sensing range to (250 + 440) = 690m.

CACP's HPBMs need to be sent at a sufficiently high power such that they may be received by all nodes within carrier-sensing distance. The power required for reaching a distance of 690m in NS-2 is 16.6035W, as compared to 0.2818W for reaching the regular reception range of 250m. The significant increase is due to the fact that transmission power grows hyperbolically with increasing distance.

The delay between the HPBM and RREP at each hop is 20ms, while that between the PRPM and RREP at the destination is 30ms. The maximum size of the PRPM is 20 bytes. These values were obtained experimentally. We omit the details of these experiments due to lack of space.

CBR is used as the traffic application, with the data packet size set to 512 bytes. The bandwidth and duration of the data sessions is varied in different experiments. Since our data sessions are uni-directional, we do not include the destination node in the CC calculation.

5.2. Performance Metrics

We compare the protocols based on the following performance metrics:

- **CC error**: This is the average difference between the CC obtained by the protocol being tested and that obtained by the Ideal method described in Section 5. The lower the error, the more accurate the protocol.
- **CC latency**: This is the average delay incurred in calculating the CC from the start of the route discovery procedure. A high value of CC latency increases the delay experienced by application flows waiting for admission into the network.
- **Number of CC calculations**: This is the average number of CC calculations performed by each node before the final CC value is obtained. Fewer CC calculations are preferred for simplicity and efficiency.
- **Number of control packets transmitted**: It is desirable to reduce the number of control packet transmissions, since they cost energy and consume network bandwidth. The lower the number of transmissions, the greater the efficiency of the protocol. The size of the control packets is ignored in this metric.
- **Number of control bytes transmitted**: This metric is similar to the previous metric, except that here we measure control bytes transmitted rather than control packets. Again, a low value for this metric is desirable.
- **Number of control packets processed per node**: Since reception and processing of control packets costs

energy, it is desirable for nodes to receive and process a small number of control packets.

- **Route acquisition latency**: This is the time interval between the initiation of route discovery by a source and the corresponding receipt of a route reply. A lower route acquisition latency corresponds to a faster response time to the application.
- **Data packet delivery fraction**: This is the fraction of data packets sent by a source node that reach the destination. If the overhead imposed by a protocol is too high, it interferes with the network's ability to deliver packets to their destinations. This metric is thus a measure of the effectiveness of the protocol in enabling successful data packet delivery. A high value for this metric is desirable.

5.3. Simulation Scenarios

We use two simulation scenarios to test the protocols. In the first scenario, ten nodes are arranged in a simple topology in order to observe the performance of the protocols in a deterministic environment. The second scenario consists of 50 nodes placed in random topologies.

We do not consider node mobility in our experiments. On-demand routing protocols assume that the topology of a network is fairly static during route discovery. As the CC determination is a part of route discovery, we can make the same assumption. Under this assumption, mobility does not significantly affect CC determination. Therefore, for simplicity, we only simulate static topologies in this paper. Note that mobility could cause the CC of a flow to change after the flow has been admitted, and this could impact the QoS of the flow. Hence, it may be beneficial to continuously monitor the QoS, and re-evaluate the admission decision when necessary, in a mobile environment.

Line topology: Our first simulation scenario is illustrated in Figure 7 and consists of ten nodes placed in two parallel lines. The distance between the lines is greater than the reception range of the nodes, so nodes from one line cannot communicate with those from the other. However, the nodes from the two lines are within carrier-sensing range of each other, and therefore they contend for medium access.

Two CBR data sessions are created. The CC determination protocol comes into play at the start of each data session. The first session, between nodes 5 and 9, acts as the background session; its purpose is to generate load in the network. We vary the bandwidth of this session from 20 to 100 Kbps in order to observe the performance of the protocols under different amounts of network load. The second data session, between nodes 0 and 4, starts after the first session has commenced. The CC determination protocols are evaluated during the start of the second session. Since the two sessions contend for medium access, the performance of the CC protocols is impacted by the load created by the background session.

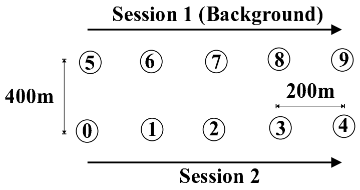


Figure 7. Line topology.

Random topology: The second simulation scenario consists of 50 nodes randomly placed in a 1500m x 650m area. Results are averaged over ten different random topologies. One to five background data sessions of 20 Kbps each are created in each simulation. Consequently, the network load varies with the number of sessions. After all the background sessions have been established, a new data session is started. The performance of the CC protocols is evaluated at the start of this last data session. The CC protocol performance is impacted by the level of network load created by the background sessions.

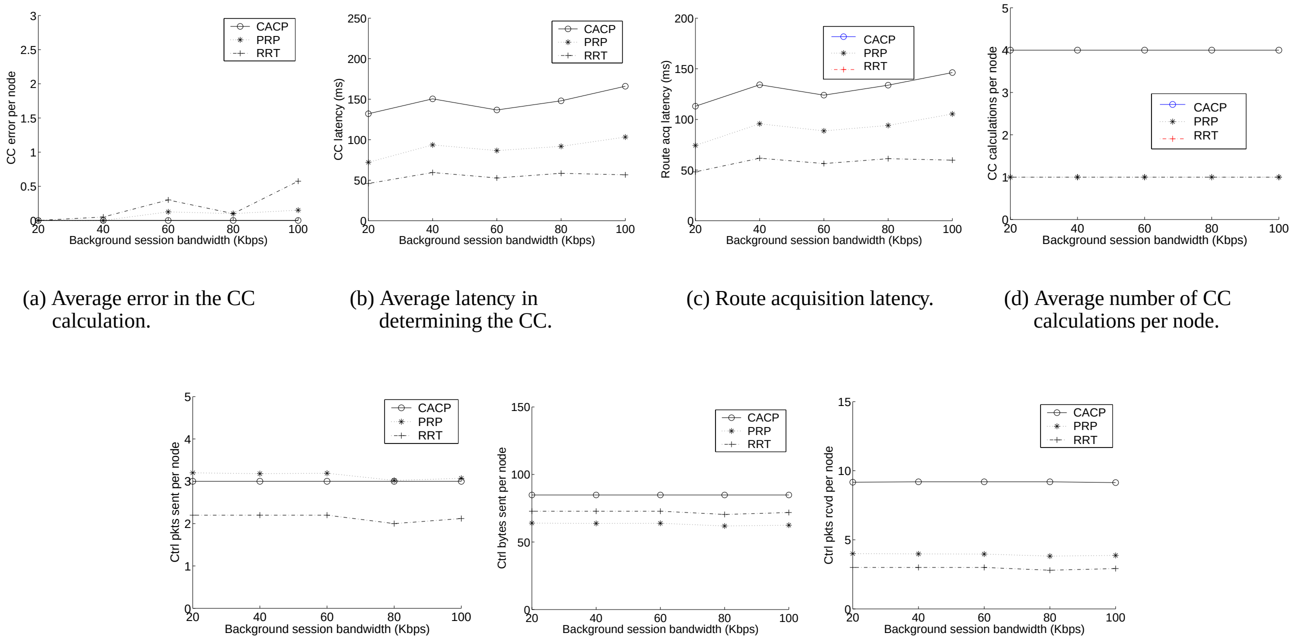
5.4. Simulation Results

Figure 8 shows the results from the first simulation scenario. Each data point is averaged over 10 simulation runs with the random number generator seeded differently in each run. We do not plot the data packet delivery fraction for this scenario since it is 100% for all the protocols.

As seen in Figure 8(a), CACP performs an accurate CC calculation in this simple scenario. Both PRP and RRT have non-zero error as the network load increases. The ability of a node to correctly sense the duration of transmissions from its CSN is adversely affected by collisions. The frequency of collisions increases with network load and so the accuracy of PRP and RRT diminishes. However, the maximum error is less than 0.7 in this scenario.

Figure 8(b) shows that CACP has the highest CC latency due to a per-hop delay between the transmission of the HPBM and the RREP. Also, in CACP, unlike PRP and RRT, the correct CC need not be known when the RREP is processed, and can change as HPBMs are received from nodes further along the path. RRT has the lowest latency since it involves no extra delays for the CC calculation. The latency of PRP is a little higher than RRT due to the extra delay injected by the destination between the transmission of the PRPM and the RREP. The latency is significantly lower than that of CACP. The route acquisition latency of the protocols, as shown in Figure 8(c), is affected by similar reasons. It is lowest for RRT and highest for CACP.

The number of CC calculations performed by each node is presented in Figure 8(d). Both PRP and RRT compute the CC only once when processing the RREP. CACP, on the other hand, must recalculate the CC after each HPBM is received, and so the average number of calculations is equal to the CC.



(a) Average error in the CC calculation.

(b) Average latency in determining the CC.

(c) Route acquisition latency.

(d) Average number of CC calculations per node.

(e) Number of control packets sent.

(f) Number of control bytes sent.

(g) Number of control packets received.

Figure 8. Performance results for line topology.

As seen in Figure 8(e), CACP and PRP transmit a higher number of control packets than RRT; this is due to the transmission of the HPBM and PRPM, respectively, at each hop along the path. We note that HPBMs are sent at a higher transmit power and therefore consume more energy than the PRPMs. RRT transmits the lowest number of control packets since no additional packets are generated other than those required by regular route discovery. Figure 8(f) shows the byte overhead. Since CACP has path accumulation on the AODV packets, plus extra control packets containing the identities all the nodes on the route, its byte overhead is highest. The byte overhead of RRT is next, since each RREQ packet is extended with a tail. PRP has the lowest byte overhead since the RREQ/RREP packets do not carry any extra information. In PRP, there is an additional control message; however, this message is fairly small in size and is only transmitted along the selected path. This is unlike the RREQs that are transmitted throughout the network.

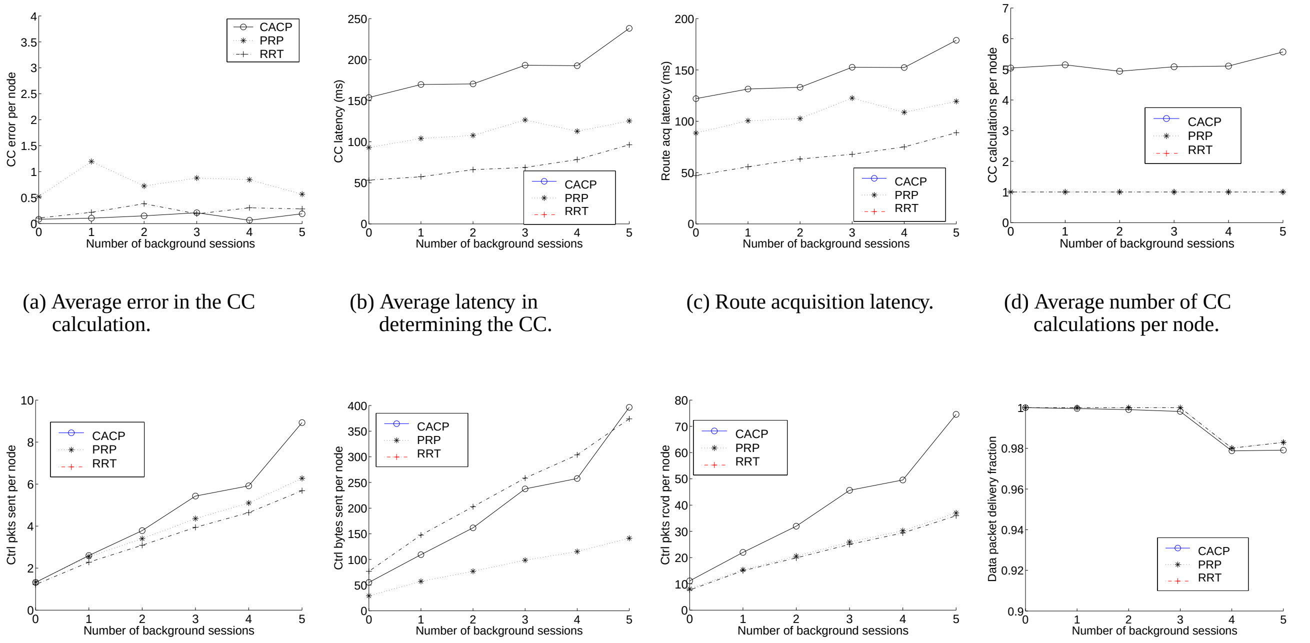
In Figure 8(g), we observe that the average number of control packets processed per node is significantly higher for CACP since the HPBMs are received and processed by all nodes within carrier-sensing range. PRP and RRT are far more efficient in this regard. The number of control packets processed is slightly higher for PRP than for RRT because of the extra PRPM transmissions.

Figure 9 presents the results from the random topology simulations. As seen in Figure 9(a), CACP shows non-zero CC error in this scenario due to the occasional collision of HPBMs with other packets. The error is still higher for PRP and RRT since these methods rely on carrier-sensing information and are therefore affected more significantly by collisions. The maximum error, however, is only about one.

Figures 9(b), 9(c) and 9(d) show the average CC latency, route acquisition latency and number of CC calculations, respectively. These graphs all follow the same trends as the previous scenario for the same reasons as described earlier.

The number of control bytes transmitted by RRT is higher than CACP in this scenario, as seen in Figure 9(f). With more nodes in the network, there are more RREQ transmissions, and so the effect of the RREQ tail exceeds that of the additional control messages in CACP and PRT. CACP's overhead is still higher than that of PRP due to the larger size of the HPBM messages. The number of control packets transmitted and processed by each node, as seen in Figures 9(e) and 9(g), respectively, increase with the increasing number of background sessions due to the greater number of route discoveries performed. The relative trends of the three protocols in these figures are the same as in the previous simulation scenario for the same reasons.

Finally, in Figure 9(h) we observe that the data packet delivery fraction is slightly lower when using CACP compared to the other protocols. This is because CACP's high power messages impact other transmissions in a larger area and increase the number of collisions. As route discoveries are performed more frequently, this effect becomes more pronounced and causes higher packet loss in the network.



(a) Average error in the CC calculation.

(b) Average latency in determining the CC.

(c) Route acquisition latency.

(d) Average number of CC calculations per node.

(e) Number of control packets sent.

(f) Number of control bytes sent.

(g) Number of control packets received.

(h) Data packet delivery fraction.

Figure 9. Performance results for random topology.

6. Conclusion

In this paper, we propose two new approaches to determine intra-flow contention, i.e. the number of nodes on a multihop path that contend for medium access. Our approaches, Pre-Reply Probe (PRP) and Route Request Tail (RRT), are based on the fundamental idea that carrier-sensing information, such as the duration of sensed transmissions, can be used to gather information about carrier-sensing neighbors. This idea is the central contribution of this paper. We compare our approaches with the intra-flow contention determination mechanism of the Contention-Aware Admission Control Protocol (CACP). Simulation results show that although PRP and RRT are slightly less accurate than CACP, the small error is heavily outweighed by benefits such as reduced network load, lower energy consumption and faster response time. Our future work consists of enhancement of the proposed protocols to improve accuracy, possibly using other types of carrier-sensing information.

Acknowledgment: This work is supported in part by NSF Career Award CNS-0347886.

References

[1] G.-S. Ahn, A. Campbell, A. Veres, and L.-H. Sun. SWAN: Service Differentiation in Stateless Wireless Ad hoc Networks. In *IEEE INFOCOM*, New York, NY, June 2002.

[2] I. Chakeres and E. Belding-Royer. PAC: Perceptive Admission Control for Mobile Wireless Networks. In *QShine*, Dallas, TX, October 2004.

[3] K. Fall and K. Varadhan. ns Manual. *http://www.isi.edu/nsnam/ns/doc/*, 1999.

[4] S. Gwalani, E. Belding-Royer, and C. Perkins. AODV-PA: AODV with Path Accumulation. In *IEEE ICC*, Anchorage, Alaska, May 2003.

[5] S.-B. Lee, G.-S. Ahn, X. Zhang, and A. Campbell. INSIGNIA: An IP-Based Quality of Service Framework for Mobile Ad Hoc Networks. *Journal of Parallel and Distributed Computing*, 60(4):374–406, April 2000.

[6] E. Nordstrom and B. Wiberg. The AODV-UU Implementation for NS-2. http://www.docs.uu.se/scanet/aodv.

[7] C. E. Perkins and E. M. Belding-Royer. Quality of Service for Ad hoc On-Demand Distance Vector Routing. *IETF Internet Draft, draft-ietf-manet-aodvqos-02.txt*, November 2001. (Work in Progress).

[8] C. E. Perkins and E. M. Royer. Ad-hoc On-Demand Distance Vector Routing. In *IEEE WMCSA*, February 1999.

[9] A. Woo, K. Whitehouse, F. Jiang, J. Polastre, and D. Culler. The Shadowing Phenomenon: Implications of Receiving During a Collision. Technical Report CSD-04-1313, University of California at Berkeley, March 2004.

[10] Q. Xue and A. Ganz. Ad hoc QoS on-demand routing (AQOR) in Mobile Ad hoc Networks. *Journal of Parallel and Distributed Computing*, 63:154–165, October 2002.

[11] Y. Yang and R. Kravets. Contention-Aware Admission Control for Ad Hoc Networks. Technical Report 2003-2337, University of Illinois at Urbana-Champaign, April 2003.