



Pre-Reply Probe and Route Request Tail: Approaches for Calculation of Intra-Flow Contention in Multihop Wireless Networks

KIMAYA SANZGIRI

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

IAN D. CHAKERES

Department of Electrical & Computer Engineering, University of California, Santa Barbara, CA 93106, USA

ELIZABETH M. BELDING-ROYER

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

Published online: 9 December 2005

Abstract. Several applications have been envisioned for multihop wireless networks that require different qualities of service from the network. In order to support such applications, the network must control the admission of flows. 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 forwarded by different hops along a multihop path, resulting in an increase in the actual bandwidth consumption of the flow to 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 examine methods to determine the extent of intra-flow contention along multihop paths in both reactive and proactive routing environments. The highlight of the 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.

Keywords: multihop wireless networks, admission control, intra-flow contention

1. Introduction

The easy availability, increasing capabilities and decreasing costs of wireless computing devices, together with the advancement of wireless communication technology, have made wireless multihop networks possible and practical. These networks have significant advantages over traditional wired and infrastructured wireless networks, such as quick and easy deployment, self-configuration, self-management, and no requirement for established infrastructure. These advantages make wireless multihop networks highly desirable in many deployment scenarios.

Several multimedia applications that have been envisioned for wireless multihop networks involve the streaming of real-time data. Such applications are typically sensitive to end-to-end delay and jitter and require either resource guarantees or priority service from the network. This is different from traditional applications, such as bulk data transfer, that require only store-and-forward capabilities. In order to effectively support these diverse applications, the network must be capable of offering different qualities of service (QoS) based on the needs of the application.

It is not possible to guarantee service quality and resource availability to application flows in wireless networks without controlling traffic admission into the network. To control wireless contention, the network must ensure that sufficient resources are available for a new flow before the flow is admitted. Further, the new flow should not adversely affect the service quality of other ongoing flows. Admission control is thus an important component of a network QoS solution.

In order 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 multimedia applications, we focus on estimation of bandwidth consumption of a flow in this paper. In wired networks, this problem is trivial; the bandwidth consumed by the flow is simply equal to that originally requested by the application. For example, if an application requests 16 kbps bandwidth from the network, the resulting flow will consume 16 kbps at every hop along the route.

The problem is significantly more complicated in wireless multihop networks 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. In other words, the CC at a node can be defined as the intersection of the set of nodes that lie on the multihop path with the set of nodes that lie within carrier-sensing range. 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 important to accurately estimate the bandwidth consumption of a flow, and thereby make a correct admission control decision.¹ Unfortunately, calculation of 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. As explained in Section 2.1, a node cannot directly communicate with nodes that are located outside its transmission range.

Early attempts at admission control in wireless multihop networks ignored the effect of intra-flow contention. To the best of our knowledge, only one solution has been previously proposed that correctly computes the CC [14]. This solution uses high power transmissions so that each node can reach its carrier-sensing neighbors, learn their identities and thereby calculate the CC. Unfortunately, high power transmissions require a capable radio and are expensive in terms of both energy and the number of nodes impacted by these transmissions.

In this paper, we examine two approaches to determine the CC. The first approach removes all high power transmissions. The second approach further increases performance by removal of all additional control messages and uses only existing routing control messages. 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, on-demand routing protocols, intra-flow contention and previous work. In Section 3 we describe approaches to

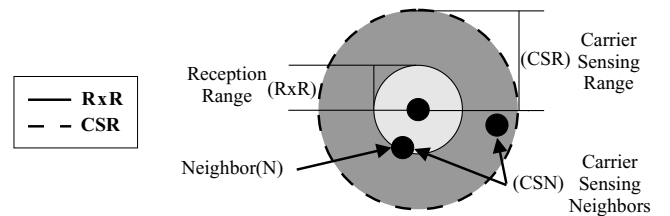


Figure 1. Notable ranges of IEEE 802.11 wireless communication. Nodes within reception range are called neighbors (N), while those within carrier-sensing range are called carrier-sensing neighbors (CSN).

determine the CC. In Sections 4 and 5 we evaluate the performance of the different techniques for calculation of the CC. 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. In Section 2.2, we briefly describe the route discovery procedure used by several popular on-demand routing protocols, such as AODV [10] and DSR [7]. Our proposed methods for calculating the intra-flow contention are integrated with the on-demand route discovery procedure. Section 2.3 explains the concept of intra-flow contention in detail, and Section 2.4 reviews previously proposed approaches for determining the CC.

2.1. Impacted area

In this section, we describe the notable ranges of IEEE 802.11 wireless communication. Knowledge of these ranges is necessary to understand the concepts of intra-flow contention and contention count.

Within a short range, wireless nodes are capable of direct communication. The maximum separation between a sender and receiver for successful packet reception is called the reception range (RxR), as shown in figure 1.² Nodes within RxR of a particular sender can directly communicate with the sender and are considered its neighbors (N).

The maximum distance at which 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

¹Note that the CC calculation is only one of the components required for a complete admission control solution for multihop wireless networks. In addition to this, an admission control solution must determine bandwidth availability and address issues such as inter-flow contention.

²While 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.

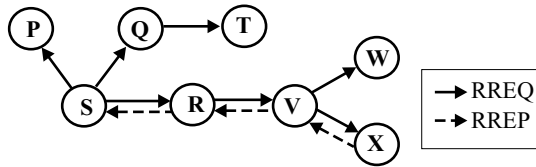


Figure 2. On-demand routing protocol route discovery.

helps avoid collisions at receivers. The larger the CSR, the fewer the collisions, at the cost of reduced spatial reuse.

2.2. On-demand route discovery

When using on-demand routing protocols, data sessions in multihop wireless networks are typically preceded by route discovery to find a route between the source and destination nodes. Since admission control needs to be performed before a session commences, admission control procedures are often integrated with route discovery.³ CC determination is required for admission control. We therefore integrate our solutions for determining the CC with the on-demand route discovery procedure. Note that our proposed solutions are conceptually independent of routing, and can also be applied to proactive routing environments with appropriate modifications, as described in Sections 3.4 and 3.5.

Most reactive routing protocols, such as AODV [11] and DSR [7], have similar route discovery mechanisms. Figure 2 shows a simple network topology. Node S, the source, wishes to communicate with node X, the destination. Node S starts the route discovery procedure by broadcasting a Route Request (RREQ) packet. Every node that receives the request re-broadcasts it until it reaches the destination.⁴ In the figure, the request is broadcast by nodes R and V, and finally reaches node X. The destination node then unicasts a Route Reply (RREP) back to the source, node S. The reply travels back on the same route that was followed by the request, and finally reaches the source, thereby completing the route discovery.

In AODV, intermediate nodes set up a reverse route to the source when they receive the request and a forward route to the destination when they receive the reply. Thus, nodes are only aware of the next hop on the route and do not know the identity of other nodes that lie on the path. In DSR, on the other hand, each node appends its address to the RREQ before re-broadcasting it further. Each instance of the request thus accumulates the address of each node it has traversed. When the request reaches the destination, the destination copies the list of addresses to the reply and unicasts it along the accumu-

³One concern here is that the route discovery flood may itself affect the service quality of ongoing sessions. This can be handled by the MAC layer by reserving a portion of the bandwidth for control traffic and restricting the control traffic to this limit.

⁴This is a generalization. In some reactive routing protocols, intermediate nodes with routes to the destination can also reply. However, if admission control is integrated with route discovery, it is often desirable that the RREQ packet travels to the destination so that resource availability may be checked along the entire route.

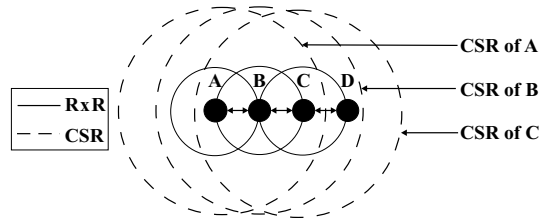


Figure 3. Example of intra-flow contention. For this path, the CC at nodes A, B and C is three.

lated path using source routing. This address accumulation may be beneficial for route caching or, as we discuss later, calculating the CC.

2.3. Calculating the contention count

As stated in Section 1, the CC 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:

$$CC_i = |CSN_i \cap NoP| + 1 \quad (1)$$

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 3 communicates with node D via nodes B and C. Each packet transmitted by node A must 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 outside of its transmission range but within its 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 [14], but this method has several drawbacks, such as consumption of additional energy and an increase in collisions. In this paper, we examine mechanisms to detect the CSN that lie on a multihop path without the use of high power transmissions.

2.4. Related work

QoS in wireless multihop networks is a popular area of research, and several QoS routing and admission control solutions have been proposed [1,8,10]. Many of these solutions completely ignore the effect of intra-flow contention. To the best of our knowledge, two approaches to determine the CC have previously been developed. In the following, each of these algorithms is described and their drawbacks are mentioned.

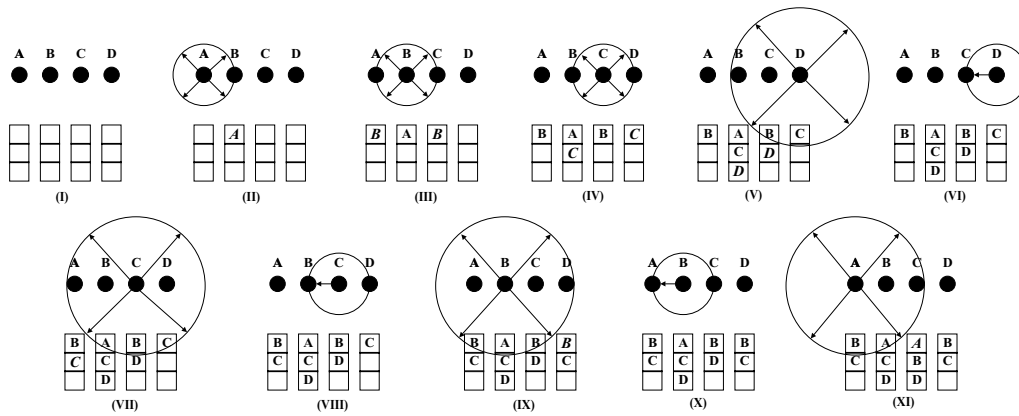


Figure 4. Contention count calculation with CACP.

Ad hoc QoS on-demand routing (AQOR): In AQOR [13], the authors correctly consider that a single wireless node cannot transmit and receive messages simultaneously. However, they ignore contention between multiple nodes that are located within carrier-sensing range. As a result, 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 3, the CC at each node is three, but AQOR computes it to be two.

Contention-aware admission control protocol (CACP): CACP [14] 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 CC, and do not consider the bandwidth availability determination or the admission control portion of the protocol.

In CACP, nodes use high power transmissions to communicate directly with their carrier-sensing neighbors. The example in figure 4 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 in step (II) of the figure. Nodes B and C re-broadcast the request, appending their node ID prior to transmission, as seen in steps (III) and (IV). 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 Powered Broadcast Message (HPBM) at a power high enough that all CSN can successfully receive the message.⁵ This is illustrated

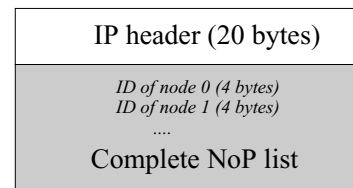


Figure 5. HPBM packet format. The shaded region is variable in size.

in step (V) of the figure. The HPBM is received by all CSN of node D and contains the NoP list (the list of nodes the RREQ traversed, including the source and destination, that was accumulated in the packet), as shown in figure 5. 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, on reception of the HPBM from node D, node C calculates the CC using equation (1). In this case:

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

After sending the HPBM, the destination transmits the RREP message that contains the NoP list to the next hop toward the source, as shown in step (VI). 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):

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

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 commu-

⁵CACP assumes that the carrier-sensing range is double the transmission range. This is not necessarily true. The carrier-sensing range depends on

the transmission power, receiver sensitivity and propagation characteristics of the wireless medium.

nicate 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. In other words, if the CC of a node is N , the node performs the CC calculation N times. 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.⁶ 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 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 and makes the packets more prone to collisions.

In the following section, we describe two 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 describe approaches for determining the intra-flow contention count. The fundamental idea behind our approaches is to use carrier-sensing information to infer the CSN of each node. Through channel measurements and control packet information, nodes can infer their CSN from regular-powered transmissions and do not need to increase their transmission power. In Section 3.1, we first describe how carrier sensing is performed and how we can use it to infer the needed information. Then, in Sections 3.2 and 3.3, we describe our two approaches. Like CACP, our proposed approaches are integrated with the route discovery procedure of reactive routing protocols. However, the basic idea is independent of routing and can also be applied to proactive

⁶The CC must be known in order to estimate the bandwidth consumption of the new flow, as described in Section 1. To control admission, the node then compares the estimated bandwidth consumption with its available bandwidth to decide whether the new flow can be admitted.

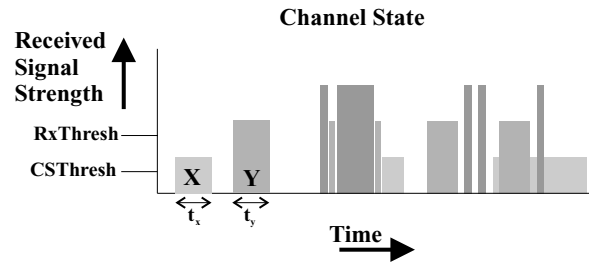


Figure 6. Diagram of received signal strength versus time. Signal strength is utilized to determine the length of packets transmitted by nodes within carrier-sensing range.

routing environments with appropriate modifications. This is briefly described in Sections 3.4 and 3.5.

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, though they may not be able to decode the contents of the packet if they are outside reception range of the sender. The ability of a packet to be decoded depends upon its received signal strength, which varies at each receiver and is affected by the distance from the sender and other factors. We assume that the value of the received signal strength is provided by the hardware.

Figure 6 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 of the receiving node, 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} .

If a node is transmitting a packet, the received signal strength of its own transmission is extremely high and drowns out the received signals from any simultaneous transmissions of other nodes. Therefore, a node cannot simultaneously transmit and receive packets.

Given the signal strength measurements, a node can construct a graph showing the channel state over time, similar to the one in figure 6. From this graph, it can determine the length of packets. For example, in figure 6, 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 measure-

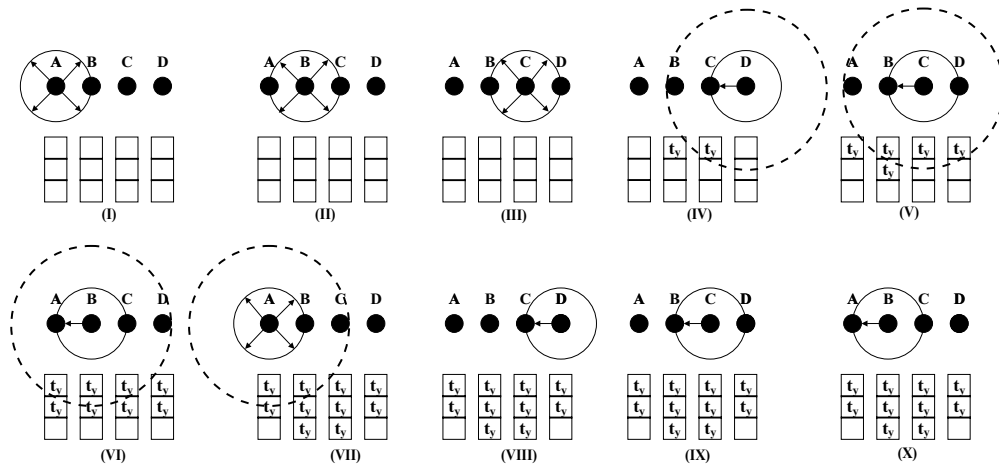


Figure 7. Contention count calculation with PRP.

ment. The ability to correctly decode 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 [12]. For example, suppose the received signal strengths of two packets are P_x and P_y . A node can capture packet X if $P_x/P_y > C_{\text{thresh}}$. Similarly, if $P_y/P_x > C_{\text{thresh}}$, packet Y can be captured. If neither condition is true, neither packet X nor Y are receivable or decodable.

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 stored by the node in a table, which we call the *carrier-sensing table*, and is soft-state, i.e. it is deleted after some time interval. The operation of PRP can be described through the example illustrated in figure 7. 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, as shown in step (I) of the figure. Steps (II) and (III) show that the request is rebroadcast by each intermediate node as in the route discovery procedure described in Section 2.2. No additional processing is required during this phase.

When the destination receives the RREQ, it generates a Pre-Reply Probe Message (PRPM) as shown in step (IV). The complete packet format is shown in figure 8. 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

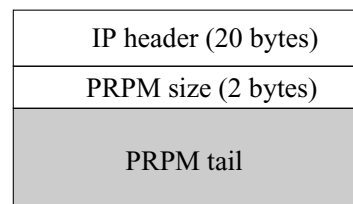


Figure 8. PRPM packet format. The shaded region is variable in size.

destination. These nodes then add the value t_y to their carrier-sensing tables. Referring to step (IV) in figure 7, the PRPM transmission by node D is sensed by nodes B and C, and both nodes record the duration t_y in their tables.

Upon reception of a PRPM, intermediate nodes process the message by forwarding it to the next hop toward the source as shown in steps (V) and (VI). The time duration of each transmission of the PRPM is recorded by all nodes located within carrier-sensing range of the sender. In step (V) of figure 7, 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, as shown in step (VII). 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. This is shown in step (VIII) of the figure. 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, in step (VIII), node C knows that it heard a packet of size t_y transmitted three times, and it also transmitted the

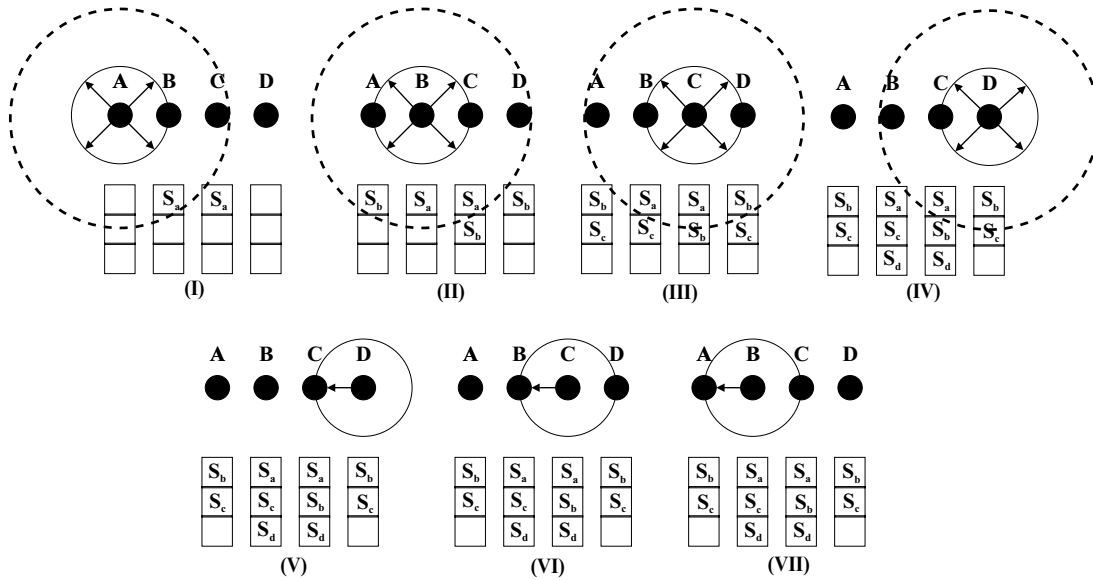


Figure 9. Contention count calculation with RRT.

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. In steps (IX) and (X), nodes C and B forward the PRPM to the source after processing it and determining their CC.

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

IP header (20 bytes)
RREQ information (32 bytes)
<i>RREQ size at node 0 (4 bytes)</i> <i>RREQ size at node 1 (4 bytes)</i>
Accumulated list of RREQ sizes
Tail appended by last node

Figure 10. RREQ packet format in RRT. The shaded region is variable in size.

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 9, 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 complete packet format is shown in figure 10. The source then broadcasts the RREQ as shown in step (I) of figure 9. Nodes B and C, which lie within node A's carrier-sensing range, record the size of the RREQ packet.

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 step (II) of figure 9, 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. The same procedure is repeated at node C in step (III).

When the destination receives the RREQ, it repeats the procedure followed by the intermediate nodes and rebroadcasts the RREQ one more time as shown in step (IV). 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 as seen in step (V).

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 step (VI) of figure 9, node B has seen packets of size S_a, S_b, S_c and S_d . Therefore, it calculates its CC to be four. Node B then forwards the RREP to node A as shown in step (VII).

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), particularly if the 16 byte node IDs in IPv6 are used. 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 durations, and are therefore 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.

3.4. Extension to proactive routing environments

The approaches described in the previous sections, PRP and RRT, are integrated with the route discovery procedure of a reactive routing protocol. However, as mentioned earlier, the basic idea of using carrier-sensing information to detect carrier-sensing neighbors and determine the contention count is independent of the routing mechanism. In this section, we briefly describe how the idea may be applied to a proactive routing environment.

Proactive routing protocols discover and maintain routes to all destinations in the network at all times, irrespective

of whether a route to a particular destination is currently required. In this respect, they are similar to traditional routing protocols for wired networks. Popular proactive routing protocols for multihop wireless networks, such as Optimized Link State Routing (OLSR) [4] and Topology Broadcast based on Reverse-Path Forwarding (TBRPF) [2], are based on the link-state routing methodology. We describe how our idea may be applied to a generic link-state routing protocol for multihop wireless networks.

In link-state routing protocols, each node determines its reachability to one-hop neighbors. This is called the node's *link state*. The link state typically includes the list of directly reachable neighbors and the cost of reaching each neighbor. The link state of every node is then reliably disseminated throughout the network. The aggregation of link state information from all nodes in the network enables each node to construct a map of the entire network. The node can then use this map to determine the shortest route to every destination. Since link state information is distributed in a reliable fashion, all nodes are guaranteed to synchronize to an identical view of the network, thereby preventing routing loops.

The idea of using carrier-sensing information to determine the contention count can be applied to proactive link-state routing protocols in the following manner. As in PRP and RRT, each node measures the durations and records the lengths of sensed packets in its carrier-sensing table. Before broadcasting its own link state information to its neighbors, each node affixes a random-sized tail to the link state packet. This results in a unique packet length for that node. The unique length is included with the link state information, and thereby disseminated to the entire network. Note that the random-sized tail is attached only when a node broadcasts its own link state information. When broadcasting the link state information of other nodes for distribution in the network, the node ensures that the tail is removed. Information about the original unique length, however, is retained in the packet.

On receiving link state information from all nodes in the network, each node knows the unique packet length associated with every other node. It can then determine its carrier-sensing neighbors by looking in its carrier-sensing table to determine which of the unique packet lengths it was able to sense. Then, to determine the contention count for a route, the node simply compares its list of carrier-sensing neighbors to the list of nodes on the route (it can determine this list from the network map). The intersection of the two lists gives the set of contending nodes. The number of such nodes is the contention count.

In the next section, we briefly describe how our idea can be applied without using the routing protocol.

3.5. Contention count determination independent of routing

The approaches proposed in this paper, PRP and RRT, are integrated with the route discovery procedure of reactive routing protocols. As stated earlier, this makes sense since a data session is typically preceded by route discovery in a reactive

routing environment. However, this is not guaranteed to be the case. It is possible that a node already possesses a route to the destination of interest. Nevertheless, the new data session still needs to be subjected to admission control, and so the contention count must be determined. In this section, we briefly describe how our idea may be applied to such a situation.

The proposed solution for this situation is similar to the PRP approach described in Section 3.2. Once again, nodes measure and record lengths of sensed packets in their carrier-sensing tables. Before admitting the new session, the source node generates a random-sized message, similar to the PRPM, and sends it to the destination along the multihop path. Intermediate nodes forward the message without any modification. On receiving this message, the destination sends a reply back to the source. The reply contains the unique length of the previous message. When forwarding the reply, each intermediate node examines its carrier-sensing table to determine how many messages of the given length it was able to sense. This number gives the node's contention count for that route. The contention count is then used by each node to make an admission control decision.

We have thus described how our mechanism to determine the contention count can be applied in different routing scenarios. In the following sections, we present analytical and simulation-based comparisons of CACP, PRP and RRT.

4. Analytical comparison

In this section, we present a simple analytical comparison of the three protocols: CACP, PRP and RRT. We compare the protocols on the number and size of control packets transmitted, the number of CC calculations performed and the additional delay incurred in determining the correct CC value and thereby completing 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 lengths of the tails appended by nodes on the path. 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. RREQs in PRP do not carry any additional information, so there is no increase in size. 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, and hence the size increases by J .

Next, we look at the size of additional control messages used by each protocol. 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 the CC calculation and 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 over that incurred by regular route discovery. Finally, in CACP, each node calculates the CC K times, where K is the final CC value, since the CC calculation must be repeated each time an HPBM is received from a CSN on the path. In

Table 1
Contention count calculation overhead.

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 node ID

J = Size of short integer

S = Size of PRPM (random)

T = Size of RREQ tail in RRT (derived from node id)

$D1$ = 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

both PRP and RRT, the CC is calculated just once at each node when the RREP is processed.

Since both CACP and PRP introduce additional control packets, we expect their routing load to be increased. This increase should be greater for CACP due to its larger control packet. Also, both CACP and RRT increase the size of the RREQ packet. We expect that the effect of this increase in size on the routing load will dominate that of the additional control packets in large networks since RREQ packets are flooded throughout the network while the additional control packets are unicast only along the route. Further, CACP is likely to have the largest route acquisition latency since it introduces an additional delay proportional to the length of the path, while RRT's route acquisition latency is likely to be the smallest. Our simulation results, presented in the next section, justify our expectations.

5. Simulation-based evaluation

We compare the accuracy and overhead of the three protocols using simulation. The NS-2 simulator 9 is used for this purpose. We implement the three mechanisms by making appropriate extensions to the AODV-UU NS-2 implementation 9 of the AODV routing protocol. AODV path accumulation 6, 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 CC 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

Table 2 presents the simulation parameters and their default values. To prevent collisions of received packets, the carrier-sensing range should be set to $(R \times R + RID)$ [3], where $R \times R$ 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 440 m. We therefore set the carrier-sensing range to $(250 + 440) = 690$ m.

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

Table 2
Default simulation parameters.

Parameter	Value
Simulator	NS-2
Propagation model	Two ray ground
MAC protocol	IEEE 802.11
Reception range (RxR)	250 meters
Capture threshold	10.0
Receiver interference distance (RID)	440 meters
Carrier-sensing range (CSR)	690 meters
Regular transmission power	0.2818 W
High transmission power (for CACP)	16.6035 W
Delay between HPBM and RREP for CACP	20 ms
Delay between PRPM and RREP for PRP	30 ms
Maximum size of PRPM	20 bytes
Traffic type	CBR
CBR packet size	512 bytes

distance of 690 m in NS-2 is 16.6035 W, as compared to 0.2818 W for reaching the regular reception range of 250 m. The significant increase is due to the fact that transmission power grows hyperbolically with increasing distance.

The values for HPBM and PRPM delays, as well as PRPM size, 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. Further details are provided in Section 5.3. Since our data sessions are unidirectional, 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. This metric indicates the accuracy of each protocol in computing the CC. 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 quick determination of CC is important for a timely admission control decision. 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 is obtained. Fewer CC calculations are preferred for simplicity and efficiency.
- *Number of control packets transmitted*: This is a measure of the overhead imposed on the network by the protocol.

Each transmission of a control packet by an intermediate node is counted as one transmission. The size of the control packets is ignored in this metric. 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 lower the overhead and greater the efficiency of the protocol.

- *Number of control bytes transmitted*: This is another metric for measuring the network overhead imposed by the protocol. It is similar to the previous metric, except that here we measure control bytes transmitted rather than control packets, i.e. the size of control packets is taken into account. Again, a low value for this metric is desirable.
- *Number of control packets processed per node*: This metric indicates the efficiency of the protocol. Reception and processing of control packets costs energy. It is therefore 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 simple 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 for the duration of the route discovery procedure. Since 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 11 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

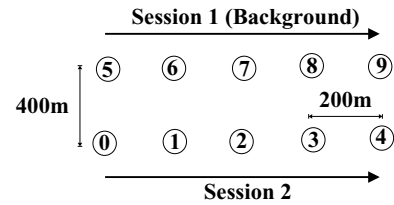


Figure 11. Line topology.

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 in this experiment. The CC determination protocol comes into play during the route discovery performed 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 discovered a route and commenced data packet transmissions. 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.

Random topology: The second simulation scenario consists of 50 nodes randomly placed in a 1500×650 m 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. Since the background sessions are in progress while the last session is established, the CC protocol performance is impacted by the level of network load created by the background sessions.

5.4. Simulation results

Figure 12 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 12(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 correct operation of these protocols depends on the ability of a node to correctly sense the duration of transmissions from its carrier-sensing neighbors. If multiple transmissions collide at a node, the node may not be able to distinguish between them, resulting in an erroneous measurement. As network load increases, collisions occur more frequently and so the accuracy of PRP

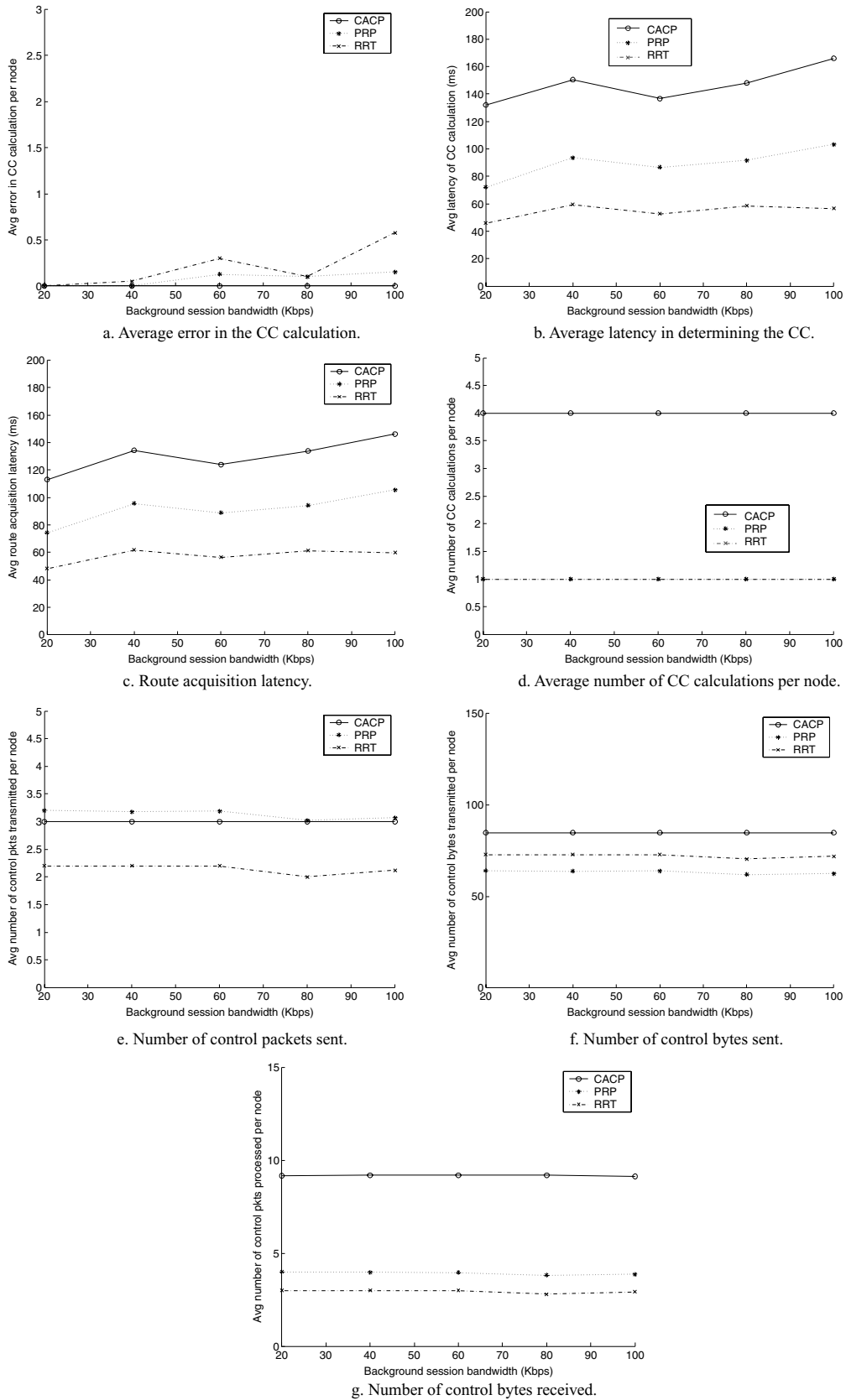


Figure 12. Performance results for line topology.

and RRT diminishes. However, the maximum error is less than 0.7 in this scenario.

Figure 12(b) depicts the average latency of the CC determination. CACP has the highest latency since it involves a delay at each hop between the broadcast of the HPBM and the retransmission of the RREP. Also, in CACP, the correct CC is not necessarily 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. Also, the final CC is known when the RREP is processed. 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 12(c), is affected by similar reasons. The latency is lowest for RRT and highest for CACP.

The number of CC calculations performed by each node is presented in figure 12(d). Both PRP and RRT compute the CC only 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.

Figure 12(e) and (f) indicate the network overhead imposed by each protocol. As seen in figure 12(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 12(f) shows the byte overhead. Since CACP has path accumulation on the AODV packets, plus extra control packets containing the identities of 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 12(g), we observe the average number of control packets processed by each node for a single route discovery. This number 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 13 presents the results from the random topology simulations. As seen in figure 13, CACP shows non-zero error in the CC calculation 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.

Figure 13(b), (c) and (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 simulation scenario for the same reasons as described earlier.

The network load of the protocols is shown in figure 13(e), (f) and (g), respectively. In this larger topology, the number of control bytes transmitted by RRT is higher than CACP, as seen in figure 13(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 PRP. 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 figure 13(e) and (g), respectively, increases 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 13(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.

5.5. Observations

The simulation results presented in the previous section indicate that the main strengths of the proposed protocols, PRP and RRT, are low network overhead and reduced resource consumption as compared to CACP. These approaches also facilitate a quick admission control decision by requiring a lower latency to determine the contention count. Their weakness, however, is that the contention count is not always determined accurately (although the error is low). An error in determination of contention count leads to an error in estimation of the bandwidth consumption of the flow, which in turn can cause an incorrect admission control decision.

From these characteristics, it is clear that PRP and RRT are most applicable in scenarios where an occasional error in admission control can be tolerated in order to reduce resource consumption and improve response time. This is likely to be the case for a large percentage of networks; many applications are not mission-critical and can tolerate occasional errors, while most wireless networks are resource constrained and low resource consumption is highly desirable. For mission-critical applications where no error in admission control can be tolerated, CACP may be more applicable, although it too produces small errors in larger networks. The disadvantages of using CACP are higher energy consumption, greater overhead and slower response time.

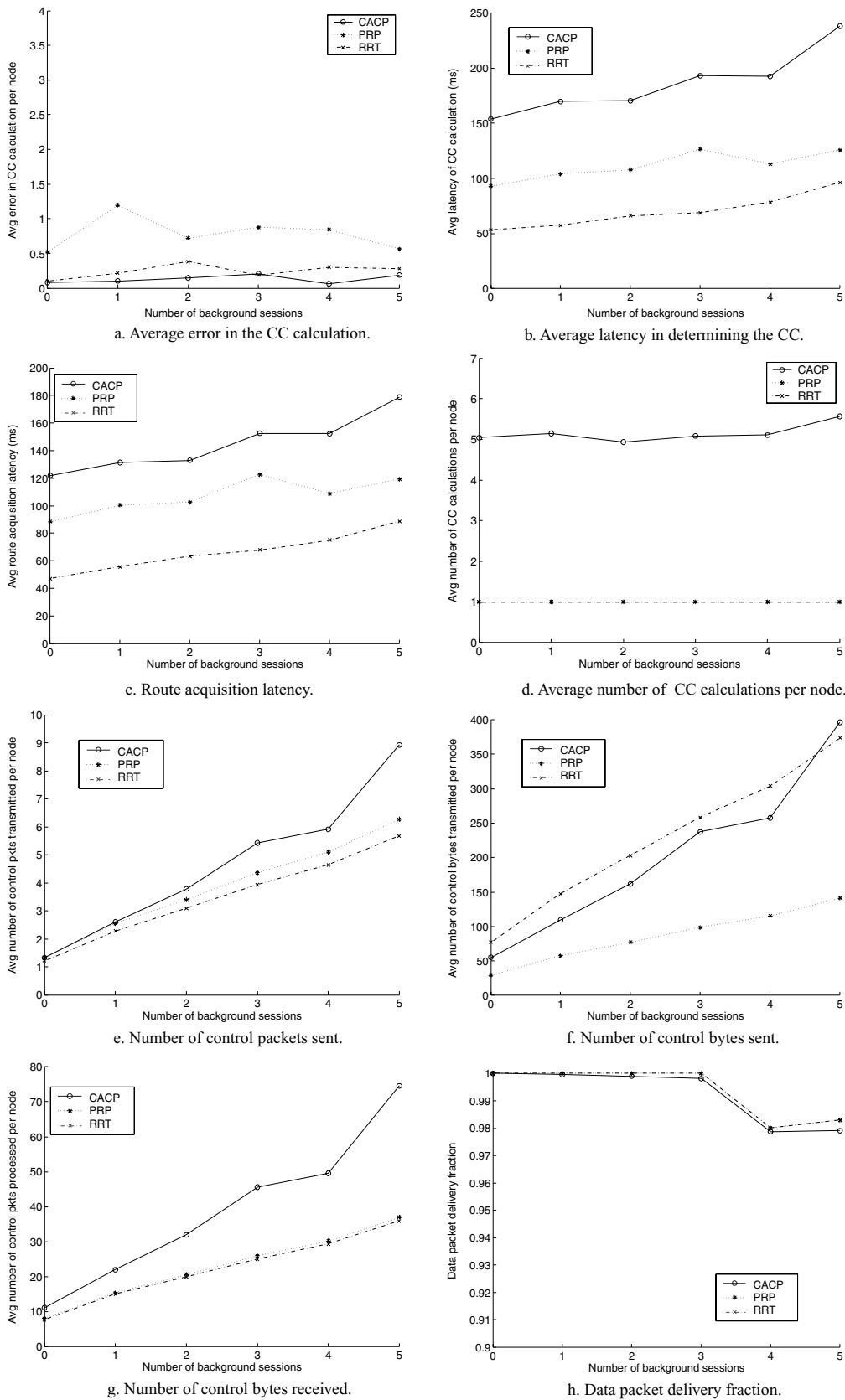


Figure 13. Performance results for random topology.

6. Conclusion

Nodes on a multihop path contend with each other for access to the shared wireless medium. This leads to intra-flow contention, i.e. contention among packets that belong to a single flow along a multihop path. The bandwidth requirements of the flow correspondingly increase. Determination of the number of contending nodes is important in order to properly estimate the bandwidth requirements of a flow and make a correct admission control decision. In this paper, we propose two new approaches to determine the number of other nodes on a multihop path that contend with a given node for medium access. This number is called the contention count. 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 Contention Aware Admission Control Protocol (CACP), which uses high power transmissions to enable nodes to communicate with their carrier-sensing neighbors. The results of our simulations show that although PRP and RRT are slightly less accurate than CACP in determining the contention count, the small error is heavily outweighed by benefits such as reduced network load, lower energy consumption and faster response time.

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 (2002).
- [2] B. Bellur and R. Ogier, A reliable, efficient topology broadcast protocol for dynamic networks, in: *Proceedings of INFOCOM*, New York, NY (1999) pp. 178–186.
- [3] I. Chakeres and E. Belding-Royer, PAC: Perceptive admission control for mobile wireless networks, in: *QShine*, Dallas, TX (2004).
- [4] T. Clausen, P. Jacquet, A. Laouiti, P. Muhlethaler, A. Qayyum and L. Viennot, Optimized link state routing protocol, in: *Proceedings of the IEEE INMIC Pakistan* (2001).
- [5] K. Fall and K. Varadhan, ns Manual (1999) <http://www.isi.edu/nsnam/ns/doc/>.
- [6] S. Gwalani, E. Belding-Royer and C. Perkins, AODV-PA: AODV with Path Accumulation, in: *IEEE ICC*, Anchorage, Alaska (2003).
- [7] D.B. Johnson, D.A. Maltz and Y.-C. Hu, The dynamic source routing protocol for mobile ad hoc networks (DSR), IETF Internet Draft, draft-ietf-manet-dsr-09.txt (2003) (Work in Progress).
- [8] 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) (2000) 374–406.
- [9] E. Nordstrom and B. Wiberg, The AODV-UU Implementation for NS-2 (2004) <http://www.docs.uu.se/scanet/aodv>.
- [10] 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. (2001) (Work in Progress).
- [11] C.E. Perkins, E.M. Belding-Royer and S. Das, Ad hoc on-demand distance vector (AODV) routing, *RFC 3561* (2003).
- [12] 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, (2004).
- [13] Q. Xue and A. Ganz, Ad hoc QoS on-demand routing (AQOR) in Mobile Ad hoc Networks, *Journal of Parallel and Distributed Computing* 63 (2002) 154–165.
- [14] Y. Yang and R. Kravets, Contention-Aware Admission Control for Ad Hoc Networks. Technical Report 2003–2337, University of Illinois at Urbana-Champaign (2003).



Kimaya Sanzgiri is a PhD candidate in the Department of Computer Science at the University of California, Santa Barbara. She is working with Prof. Elizabeth Belding-Royer in the Mobility Management and Networking (MOMENT) Laboratory. Kimaya received her B.E. (Hons.) in Computer Science from the Birla Institute of Technology and Science (BITS), Pilani, India in 1999. Her research interests are in the area of wireless networking, specifically mobility, quality of service support and security. See <http://www.cs.ucsb.edu/~kimaya> for more details.

E-mail: kimaya@cs.ucsb.edu



Ian D. Chakeres is a Ph.D. student in the Department of Electrical and Computer Engineering at the University of California, Santa Barbara. He is working with Prof. Elizabeth M. Belding-Royer in the Mobility Management and Networking (MOMENT) Laboratory. He completed his B.S. and M.S. in Electrical and Computer Engineering at Ohio State University in 1998 and 1999. He is also a co-chair of the IETF MANET working group. Ian's research interests include wireless communication and mobile networking, specifically routing protocols, MAC protocols, cross-layer coordination and quality of services in mobile wireless networks. See <http://moment.cs.ucsb.edu/~idc> for further details.

E-mail: idc@engineering.ucsb.edu



Elizabeth M. Belding-Royer is an Assistant Professor in the Department of Computer Science at the University of California, Santa Barbara. She completed her Ph.D. in Electrical and Computer Engineering at UC Santa Barbara in 2000. Elizabeth's research focuses on mobile networking, specifically routing protocols, multimedia, monitoring, and advanced service support. Elizabeth is the author of numerous papers related to ad hoc networking and has served on many program committees for networking conferences. Elizabeth is the TPC Co-Chair of ACM MobiCom 2005 and IEEE SECON 2005, and is currently on the editorial board for the Elsevier Science Ad hoc Networks Journal. Elizabeth is the recipient of an NSF CAREER award, and a 2002 Technology Review 100 award, awarded to the world's top young investigators. She is a member of the IEEE, IEEE Communications Society, ACM, and ACM SIGMOBILE. See <http://www.cs.ucsb.edu/~ebelding> for further details.

E-mail: ebelding@cs.ucsb.edu