

Multi-path Admission Control for Mobile Ad hoc Networks

Anders Lindgren

Department of Computer Science and Electrical Engineering
Luleå University of Technology

Elizabeth M. Belding-Royer

Department of Computer Science
University of California, Santa Barbara

Abstract

As wireless networks become more prevalent, users will demand the same applications that are currently available in wired networks. Further, they will expect to receive a quality of service similar to that obtained in a wired network. Included in these applications are real-time applications such as voice over IP and multimedia streams. To enable the support of applications that require real-time communication in ad hoc networks, congestion must be prevented so that the needed quality of service can be provided. An admission control mechanism is an essential component of the quality of service solution. Unfortunately, current admission control solutions encounter problems during mobility, often resulting in unacceptable disruptions in communication. To solve this problem, we apply multi-path routing mechanisms that maintain alternate paths to the destination and propose a new admission control protocol. We show through simulation that our solution is able to prevent communication disruptions and meet the QoS needs of applications better than previous solutions.

1. Introduction

Over the past few years, a continuing trend of increased multimedia usage in mobile networks has become apparent [7]. This has resulted in an increase in the desire to run real-time applications over wireless ad hoc networks. Real-time applications have strict quality of service (QoS) requirements. These requirements include high delivery rates of data packets and low end-to-end delays. It is important that these requirements are met by the network because otherwise the perceived quality of service that the user experiences will suffer, and the applications will, in effect, become unusable. Unfortunately, ad hoc networks often suffer from worse performance than their wired counterparts. This

can be due to the low throughput inherent to wireless networking technologies, frequent topology changes and resultant communication disruptions caused by node mobility, and high latencies stemming from long contention, transmission, and buffer delays.

Given the quality of service requirements and the constraints of ad hoc networks, it is clear that new methods to provide quality of service over ad hoc networks are needed. There have been numerous proposals to provide QoS in wireless networks, including several differentiation schemes for the MAC layer to allow prioritization of real-time traffic over best-effort traffic [1, 2, 13, 15]. While these schemes prevent other traffic from interfering with real-time traffic, it has been shown that if the amount of real-time traffic in the network is allowed to increase in an uncontrolled manner, network performance will deteriorate significantly [9]. Thus, one of the problems that has to be solved is that of preventing over-utilization of the wireless channel (resulting in reduced quality for all involved traffic). This can be accomplished through call admission control. If an admission control mechanism is in place, flows cannot be started at will; the admission control mechanism must first be consulted. This mechanism uses information about the current state of the network and the new flow to determine whether enough resources are available in the network to allow the new flow to start. A new flow should only be admitted if the QoS requirements of all flows in the network still can be met after the new flow begins.

Previous work on wireless network call admission control has provided solutions for non-mobile networks [16] and solutions that can handle some extent of mobility [5]. However, the performance of these solutions deteriorates in the presence of node mobility. This motivates the work presented in this paper. To increase the user's perceived quality of service, we apply multi-path routing techniques to improve the previously proposed solutions. These techniques enable a source to discover several alternate paths to the

destination, all of which are capable of providing the required quality of service. If the current path becomes unusable, the traffic flow can then quickly switch to one of the alternate paths without loss of performance. Our solution monitors each alternate path to ensure that it can provide the required quality of service. If conditions change and the QoS requirements cannot be met along a route, that route is removed from the cache. This ensures that stale routes are not used, which would result in degraded performance.

The rest of the paper is organized as follows. Section 2 presents background information and related work. In Section 3, our solution, Multi-path Admission Control for Mobile Ad hoc Networks (MACMAN), is presented. MACMAN is a protocol for admission control and routing in ad hoc networks that uses caching and maintenance of alternate paths. Section 4 describes the simulation scenarios and metrics used for testing our solution. In Section 5 we present the results of our simulations, and Section 6 concludes.

2. Background and Related Work

Due to the characteristics of the nodes and the wireless channel, ad hoc networks are inherently different from wired networks. Thus new solutions are needed to improve network performance. The channel access technology most commonly used for ad hoc networks is the IEEE 802.11 standard [14], which uses a CSMA/CA based mechanism to control access to the wireless channel. The use of CSMA/CA means that if a node senses an ongoing transmission from another node, it will defer from transmitting until the other transmission ceases. The carrier-sensing range, within which a node can detect a transmission from another node, is usually larger than the transmission range of the node. Because of this, the bandwidth available to a node is not only constrained by the bandwidth consumption of the node itself and its immediate neighbors, but also by transmissions of nodes within the wider carrier-sensing range (i.e. by its *Carrier Sensing Neighbors* (CSN)). The available bandwidth must be taken into account when designing admission control mechanisms for ad hoc networks. Further, contention occurs both among the nodes along the path, as well as between nodes on different paths. This contention affects the total bandwidth needed to admit a flow.

To perform admission control, a node must know the amount of bandwidth that is available so that it can determine whether there is enough to admit a new flow. The available bandwidth is commonly determined by Eq. (1), where $U \in [0, 1]$ is a measure of the channel utilization.

$$B_{available} = (1 - U) * B_{max} \quad (1)$$

Here, B_{max} is a measure of the maximum available bandwidth.

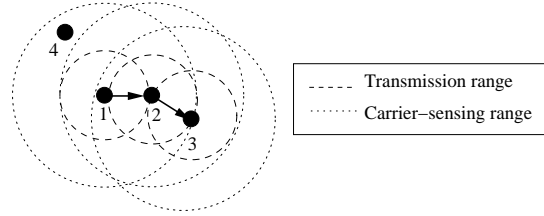


Figure 1. Problem of CACP multi-hop query method. Node 4 is within carrier-sensing range of node 1, but is not reached by the flooded query from node 1.

Previous work has shown that one of the most accurate ways to estimate the channel utilization is through *busy-time measurements* of the wireless channel [5]. This estimation method consists of measuring the amount of time the channel is sensed busy during some time window. Since the carrier sensing mechanism is used to perform the measurements, this method includes transmissions by nodes that are out of transmission range but within carrier-sensing range of the node in the bandwidth measurements. Another benefit of this method is that it is a passive method (i.e., it does not require any additional control messages to be transmitted), and therefore it can be used without the risk of reducing the available bandwidth.

2.1. Local (Single-hop) Admission Control

In this section, we discuss previous work for admission control over a single hop.

2.1.1 Contention-aware Admission Control Protocol

The Contention-aware Admission Control Protocol (CACP) [16] was specifically designed for the special properties of wireless networks. When making an admission decision, not only the locally available resources are considered, but those of nodes within carrier-sensing range are also used for the calculation. To determine the locally available bandwidth, nodes perform channel busy-time measurements and calculate the available bandwidth as in Eq. (1).

To ensure that all nodes affected by the transmission of the traffic flow have enough available resources to allow the flow to be admitted, CACP queries all nodes within carrier sensing range of the transmission path. Two methods to accomplish this are suggested by the authors – either a multi-hop approach that floods the query message to all nodes within a limited hop count, or a high power transmission for the query messages so that more nodes can be reached. The problem with the multi-hop approach is that there can be nodes within carrier sensing range that cannot

be reached through any path (as shown by node 4 in Figure 1), while the problem with the high power transmission is that it introduces more interference and collisions into the network.

When a node receives the query message, it checks its locally measured bandwidth to determine whether the flow can be admitted. If the flow should not be admitted, a message is sent back to the source to prevent the admission of the flow.

2.1.2 Perceptive Admission Control

It has been shown that the querying mechanism used by CACP can reserve bandwidth in an unnecessarily large area [5]. Further, the use of negative acknowledgments to deny flow admissions results in a large number of incorrect admission decisions, especially at high network loads.

To alleviate this, the *Perceptive Admission Control* protocol (PAC) has been proposed [5]. PAC reduces the protocol overhead and the risk of making erroneous admission decisions through the introduction of a new method to determine the available bandwidth. This method does not require the transmission of any control messages. As in CACP, the busy-time measurement is used to determine the available bandwidth. However, the main idea of PAC is to extend the range for that measurement, enabling nodes to make admission control decisions without the need for communication with other nodes. To enable this, the range that the busy time measurement covers is increased to cover the minimum distance required for two simultaneous transmissions to occur without a collision. Thus, when a node needs to make an admission control decision, it only has to check its locally available bandwidth, calculated according to the busy-time measurements. To admit a flow with a bandwidth requirement B_{req} , the inequality in Eq. (2) must be satisfied.

$$B_{available} - B_{rsv} > B_{req} \quad (2)$$

B_{rsv} is a small fraction of the bandwidth that is reserved to better handle bandwidth fluctuations and prevent the network from becoming congested. In [5], the authors discuss a number of ways to increase the sensing range, and the implementation of PAC on Berkeley Motes is described in [6].

Due to mobility, regions of the network can become congested even though admission control was initially performed. In PAC, this is handled through continuous monitoring of the available bandwidth by the nodes. If a node has an ongoing traffic flow and the available bandwidth drops below a threshold value, the flow source should stop the transmission. After a randomly selected backoff time, a node can attempt to re-admit a previously blocked flow in the hope that either more bandwidth has become available, or that another path is available. Bandwidth monitoring and flow

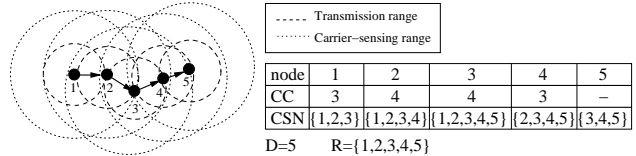


Figure 2. Intra-flow contention. Traffic is sent from node 1 to node 5. Node 1 has three nodes within carrier sensing range that transmit data for the flow (itself and nodes 2 and 3). Thus three times the flow bandwidth must be available at node 1 for it to admit the flow. The same reasoning holds for the other nodes.

pauses allow PAC to keep the network in an uncongested state, thereby enabling admitted flows to maintain their required QoS.

2.2. Multi-hop Admission Control

Two different methods of admission control over a single hop were described in the previous section. However, in ad hoc networks, it is often of interest to perform admission control decisions over a multi-hop path. In this case it is not enough to only consider the bandwidth available at a single node. There must be enough bandwidth available at every node along the path. Further, the bandwidth requirement of a flow at each node is higher than the transmission rate of the flow due to intra-flow contention [12]. Intra-flow contention occurs because bandwidth is consumed by the transmission of packets from this flow at each node on the path that is within carrier sensing range, as can be seen in Figure 2.

In [16], an on-demand source routing protocol, similar to DSR [8], that is able to perform multi-hop admission control is presented. This same approach is also leveraged by PAC [5]. The bandwidth required at each node depends on the number of nodes on the path that are within carrier sensing range of the node performing the admission decision. A flow should be admitted only if there is enough available bandwidth to support the transmission of the flow by all nodes along the path. In this approach, the contention count (CC) is defined as the multiple of the bandwidth required by the flow that must be available for the flow to be admitted. The CC is calculated by:

$$CC = |CSN \cap R \setminus \{D\}| \quad (3)$$

where CSN is the set of nodes that are within carrier sensing range of the node (including the node itself), R is the set of nodes on the route from source to destination, and D is the destination. Figure 2 shows an example of the effect

of intra-flow contention on the contention count and illustrates the CSN for each node. To calculate the contention count, a node must be aware of the identity of its carrier sensing neighbors. This can be accomplished through periodic high power transmissions that reach all nodes within carrier-sensing range.

Before performing a route discovery, the source checks to ensure that it has enough local bandwidth to support the flow itself. If enough resources are not available, it backs off and tries again later. If enough bandwidth is available, a route request is flooded through the network. The source includes the flow bandwidth requirement in the route request to enable nodes to make an admission decision. As each node receives the route request, it calculates the CC using the partial route collected in the request and performs local admission control. It uses the CC to determine whether it can support the flow by checking the following inequality:

$$B_{avail} - B_{rsv} > CC \cdot B_{req}. \quad (4)$$

If an intermediate node can support the flow, it adds its identifier to the source route collected in the request and re-broadcasts the route request. Otherwise, it drops the packet to prevent the route discovery from continuing along a path that cannot support the flow. When the request successfully reaches the destination, the destination sends a reply back to the source. The route reply contains the entire path traversed by the request. When a node receives the reply, it uses this information to make an accurate calculation of the CC . It only forwards the route reply if enough bandwidth is available to support the flow. Since nodes only know a partial route when processing the route requests, it is possible that a route that seemed acceptable when forwarding the route request is found to not be able to support the flow once the complete route is known. Thus, this check must be performed both for the route request and reply.

When the reply reaches the source, the flow has been successfully admitted along the traversed path. The source can then begin transmission of that flow immediately. Future admission decisions will now include the bandwidth of the new flow in their calculations. Because bandwidth is not explicitly reserved during the admission request, there is a small risk of two flows being simultaneously admitted even when enough resources are not available. The risk of this occurrence, however, is small, and, should it occur, this over-admission will be detected and one flow will stop.

2.3. Problems of mobility

Since mobility is inherent in most ad hoc networks, it is important that admission control schemes work properly even when nodes are mobile. Current admission control schemes do not have any suitable mechanism to handle situations where the QoS requirements can no longer be met

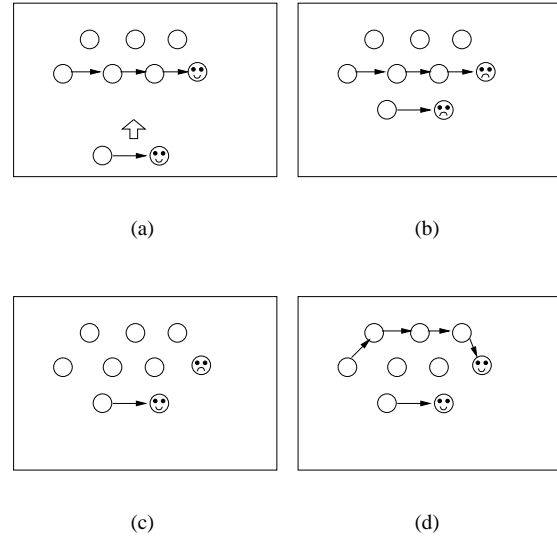


Figure 3. Problem of admission control with mobility.

due to mobility. As an example, Figure 3(a) shows two admitted flows, where each consumes all the bandwidth in its neighborhood. Later, the nodes along each path move into the vicinity of each other, and the channel becomes congested, as shown in Figure 3(b). If both flows are allowed to continue, neither will receive its QoS requirements. CACP does not have a method for handling this situation. Both flows therefore suffer from unacceptable quality of service. PAC provides a better solution by continuously monitoring the available bandwidth. If the bandwidth drops below the needed threshold during the lifetime of a flow, the transmission of that flow is stopped while the transmission of the other proceeds (Figure 3(c)). Since the exact timing of checking the channel state and restarting stopped flows is randomized, it is unlikely that both flows will detect congestion and stop at the same time. If that should happen, the randomized backoff time gives a high probability that only one of the sources will start the flow again. A similar problem occurs if one of the intermediate nodes moves away, causing one of the links in the route to break. After a randomly selected backoff time, a source can attempt to re-admit a stopped flow. However, even if bandwidth has become available or a new path is found (as in Figure 3(d)), the latency to re-admit the stopped flow causes the flow to suffer from a disruptive break in communication. Therefore, it is desirable to be able to switch directly to a new path if such an alternate path exists.

There has been some previous work that has attempted to do distributed maintenance of resource reservation paths

by performing local repairs of routes and reservations when links break [4]. The admission control mechanism presented in that work, however, does not take into consideration the effect that transmissions have on carrier sensing neighbors that are not in the transmission range of a node, and thus it is likely to be overly optimistic. Further, by waiting until a link break has been detected to try to repair the route, additional delays may occur in the system.

3. Multi-path Admission Control for Mobile Ad hoc Networks

To improve admission control performance in the presence of mobility, we propose *Multi-path Admission Control for Mobile Ad hoc Networks* (MACMAN). MACMAN builds on CACP and PAC to enable flows to meet their QoS requirements in a mobile networks. The protocol both discovers and maintains multiple paths that can provide the required quality of service between the source and the destination. This allows a source to quickly switch to an alternate path that can support the flow if the current path becomes unusable. The following sections describe the operation of MACMAN.

3.1. Route Discovery

The route discovery phase of MACMAN uses the mechanism described in Section 2.2 to discover routes that meet the QoS needs of a new flow. The method used by PAC is adopted to calculate the locally available bandwidth. Since source routing is used, it is simple to modify the route discovery process to support the use of multiple paths. Alternate routes do not require extra state in intermediate nodes since they only need to be cached at the source. When route requests reach the destination, the destination not only sends a reply to the first request received, but also replies to subsequent requests. As route replies propagate to the source, the source chooses the best route to use for transmission, where the best route is, for instance, the route with the shortest path length. It also caches the other routes for possible later use.

3.2. Route Maintenance

A problem with caching multiple routes at the time of route discovery is that, over time, the routes are likely to become stale [10]. In that case, they are no longer valid when needed either because the route has broken, or because it can no longer support the needed QoS. Previous work has shown that the transmission of periodic messages along alternate routes provides a useful mechanism for keeping track of the existence of those routes [11]. To avoid the accumulation of stale routes that no longer can provide the

required QoS, MACMAN continuously monitors each alternative route in the cache. Periodic *Route Capacity Query* (RCQ) messages are sent along each of the backup paths towards the destination. As in the route requests, the RCQ messages contain the bandwidth requirement of the flow. They also contain a listing of the route that is currently being used to transmit the flow.

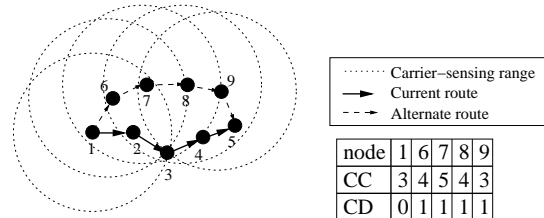


Figure 4. Calculation of Contention Difference (CD). The route 1 → 2 → 3 → 4 → 5 is currently used, and RCQ messages are sent along the alternate path 1 → 6 → 7 → 8 → 9 → 5. For example, at node 6, the normal contention count along that path is 4. However, since 3 of its CSN (nodes 1, 2, and 3) are already involved in the transmission along the current path, the contention difference is only 1. The same reasoning holds for the other nodes.

When each node along an alternate route receives an RCQ, it must determine whether it can still support the flow. To make this determination, it is important to consider that the flow is currently being transmitted along another route. If the flow was instead transmitted on this new route, the transmission along its current route would cease. Because the transmission of the flow along its current route is likely to reduce the measured available bandwidth along the alternate route, calculating the *CC* along an alternate route (as is done during the initial admission control) is not sufficient. This would result in an underestimation of the available bandwidth and the subsequent erroneous deletion of routes from the cache. To determine the increase (or decrease) of contention if the flow is transmitted on the new path, the Contention Difference (*CD*) is calculated, as shown in Eq. (5), when a node receives a RCQ message.

$$CD = |CSN \cap R_{new} \setminus \{D\}| - |CSN \cap R_{curr} \setminus \{D\}| \quad (5)$$

Here, R_{new} is the route on which the RCQ is transmitted, while R_{curr} is the route currently used for the flow. An example of *CD* calculation is shown in Figure 4. The *CD* is used to determine the bandwidth requirement of the flow

(i.e., $CD \cdot B_{req}$), and this is compared to the available bandwidth by the following inequality:

$$B_{avail} - B_{rsu} > CD \cdot B_{req} \quad (6)$$

If the node determines that the available bandwidth is **not** sufficient to meet this requirement, it sends a *Route Capacity Failed* (RCF) message back to the source of the flow. When the source receives the RCF message, it removes the route listed in the RCF from its route cache so that the route is no longer considered as a viable alternative path. On the other hand, if the node determines that the available bandwidth is sufficient to support this flow, it forwards the RCQ message to the next hop.

The propagation of the RCQ continues until the last node on the route before the destination is reached. This node is the final node to make a decision of whether the flow can be supported. Since the destination does not consume any bandwidth for this flow, it is not involved in the route maintenance process.

3.3. Route Updates Through Proactive RREQs

Through route maintenance, MACMAN ensures that routes are not kept in the cache after becoming unusable through link or QoS breaks. However, this process can only remove routes from the cache and can never add new routes. If there is mobility in the system, it is inevitable that all routes eventually will break and be removed from the cache, thus making it necessary for the flow to be stopped and a new route discovery process to be initiated. To prevent such breaks in the communication, attempts are made to discover new routes if all cached alternate routes are removed. If a source node does not have any alternate routes cached for a destination with which it is currently corresponding, it initiates a route discovery to find new routes that have become available since the last route discovery. This allows nodes to discover newly available routes before the currently used route breaks, thereby enabling a switch to the new route without the need to stop the flow to find a new route.

4. Simulation Environment

We use the NS-2 network simulator to verify the correct operation of MACMAN and evaluate its performance. We first study the specific scenario in Figure 5 to verify the operation of MACMAN. Flows of 200 kbps are started from S to D through the path $S \rightarrow M \rightarrow N \rightarrow D$ as well as between the node pairs $O \rightarrow P$ and $Q \rightarrow R$. During the simulation, the node pairs move closer to the flow from S to D , causing the wireless medium to become congested. The congestion makes it impossible for all flows to continue with maintained quality of service. Therefore, the flow from

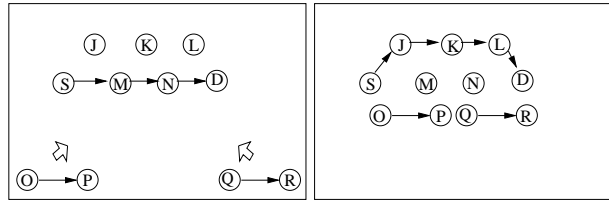


Figure 5. Scenario for initial simulations.

S to D can no longer use the current path, but must switch to the path $S \rightarrow J \rightarrow K \rightarrow L \rightarrow D$ in order for the quality of service to still be satisfied.

For a more thorough study, the second set of simulations involves a larger network with random mobility. 50 nodes are randomly placed in a $1000m \times 1000m$ area. The nodes move according to the random waypoint mobility model [3] with a pause time of 10 seconds and a variable maximum speed. 20 nodes are randomly chosen as sources of traffic to 20 other nodes. CBR traffic flows are used with a packet size of 256 bytes and a bit rate of either 32 kbps or 64 kbps. The actual amount of traffic sent in a single simulation run depends on both the traffic rate of each flow and the admission control decisions made during the simulation. If a flow cannot be admitted to the network, that source does not transmit data packets. Thus less traffic is present in the network. The first flow starts after 10 seconds, and after that, another flow starts every 5 seconds. The total simulation time is 150 seconds. All results shown are averages over 10 simulation runs. Table 1 shows the parameter settings used in the simulations.

4.1. Metrics

To evaluate the performance of the protocols, we use the following metrics that we believe reflect the important properties of the protocol:

Table 1. Simulation parameter settings.

Simulation area	$1000m \times 1000m$
Number of nodes	50
Channel bit rate	2 Mbits/s
Data packet size	256 bytes
Traffic flow data rates	32, 64 kbits/s
Traffic sources	20
Admission control backoff	$U(1, 3)s$
Pause time	10s
Max speeds	0, 1, 5, 10, 15, 20m/s
Min speed	0m/s
RCQ Interval	2s

- **Delivery rate:** This metric measures the number of packets successfully delivered during the simulation. The ability to deliver a high percentage of packets to a destination increases the overall utility of the system. Because the amount of traffic transmitted by the traffic sources varies based on the admission decisions during the simulation, it is not possible to express this metric as a percentage in a meaningful way. Therefore, we simply count the number of data packets received.
- **End-to-end delay:** The primary motivation for admission control is to support real-time traffic. Such traffic is sensitive to delay, so it is important to ensure that a new scheme does not impose large additional delays. We study the average delay for each packet to reach its destination.
- **Number of flow breaks:** Another problem that can affect real-time traffic is the blocking of an ongoing flow because the route in use becomes unusable (either due to a link break, or due to network congestion along some part of the route). Even if the flow can be resumed quickly, breaks in the flow reduce the perceived quality of the flow. Thus, we study the number of times a flow that has started has to be stopped. If there exist an alternate path that can be used instead of the now unusable path, such flow breaks can be prevented.
- **Control overhead:** Our proposed protocol introduces extra control traffic to monitor alternate paths. We study the total number of control packets sent in the network to determine the impact of this extra overhead.
- **Average number of simultaneously admitted flows:** Finally, we look at the number of flows that can be concurrently admitted. We take the measurements after the start time of the final flow, so that all flows have the chance to be admitted. We measure the average number of flows that are simultaneously active in the network.

We compare the performance of our solution with that of PAC to better understand the strengths and weaknesses of our approach.

5. Performance Results

5.1. MACMAN RCQ Interval

As a first step in our evaluation, the optimal RCQ interval for MACMAN must be determined. The RCQ interval is a tradeoff between the resiliency to path breaks and the amount of control overhead generated. RCQ transmission intervals between 1 and 10 seconds were tested. Figures 6,

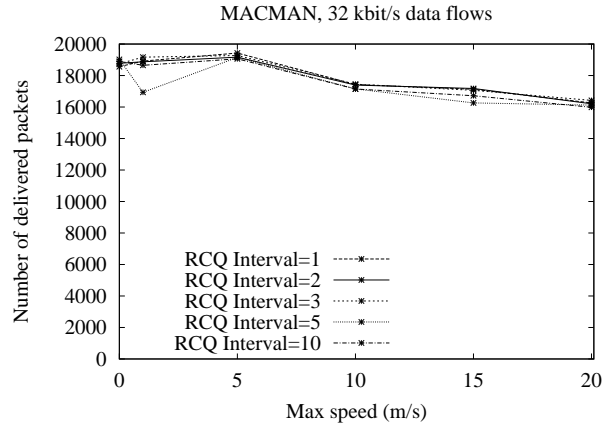


Figure 6. Average delivery rate with varying RCQ intervals for 32kbps flows.

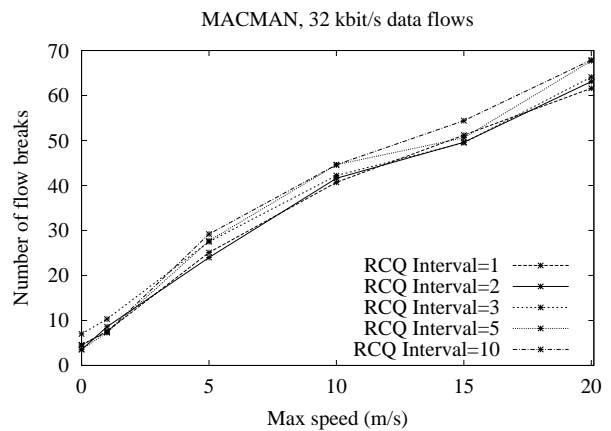


Figure 7. Average number of flow breaks with varying RCQ intervals for 32 kbps flows.

7, and 8 show the delivery rate, the number of flow breaks, and the overhead for the different RCQ intervals, respectively. The delivery rate is not notably affected by the length of the RCQ interval. However, longer intervals tend to lead to a slight increase in the number of flow breaks. As expected, the overhead imposed by the protocol decreases as the RCQ interval increases since RCQ messages and proactive route requests are sent less often. Based on the tradeoffs seen in these results, the choice was made to use an RCQ interval of 2 seconds for the remaining evaluations.

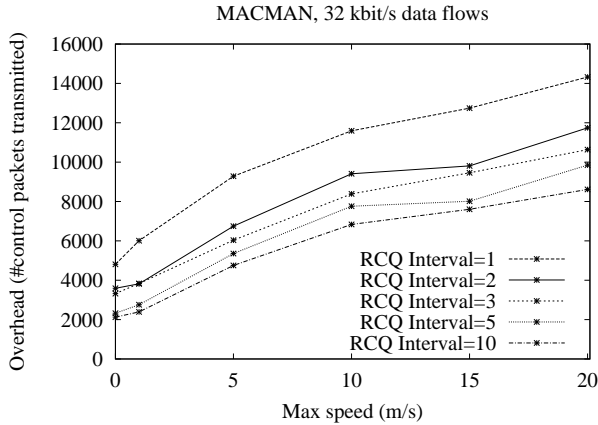


Figure 8. Control packet overhead with varying RCQ intervals for 32 kbps flows.

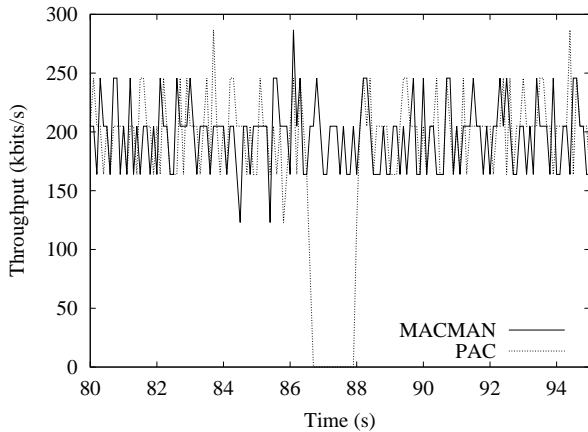


Figure 9. Single flow comparison of MACMAN and PAC.

5.2. MACMAN and PAC Comparison

We now present results comparing the performance of MACMAN and PAC. First, we compare the performance of MACMAN to the multi-hop variant of PAC using the simple scenario in Figure 5. Figure 9 shows the throughput variation over time for the flow from S to D during the portion of time when the network becomes congested. During the time before 80 seconds and after 95 seconds, the throughput pattern is similar to the 90 to 95 second time period. When mobility causes the wireless channel to become congested (just before 87 seconds into the simulation), it can be seen that the throughput of the flow suddenly drops to zero when PAC is being used. This dras-

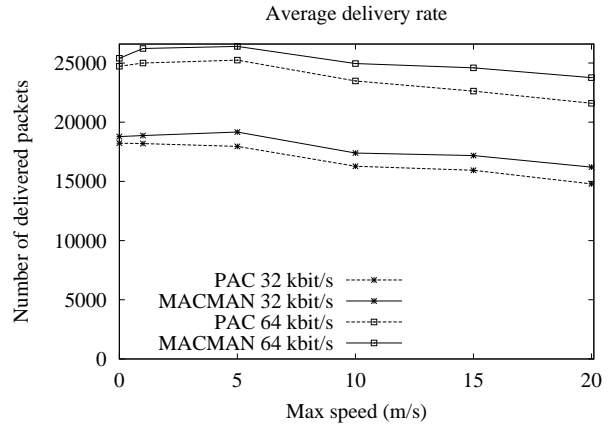


Figure 10. Average delivery rate.

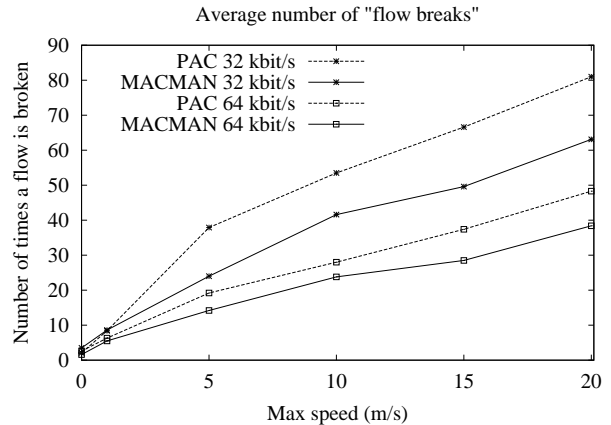


Figure 11. Average number of flow breaks.

tic change in throughput is due to the monitoring mechanism of PAC detecting that the required quality of service can no longer be met. It therefore stops the transmission of the flow. The flow does not continue until after more than a second later when a backoff and a new route discovery has occurred. This disruption in communication is not acceptable for real-time traffic. On the other hand, the figure also shows that when MACMAN is being used, no such communication disruption happens. MACMAN is able to cache an alternate route ahead of time and seamlessly switch to that route when the current route becomes congested.

After verifying that MACMAN is able to prevent communication disruptions in the smaller, more controlled scenario, we now investigate the results for the larger network setting. Figure 10 shows the average number of packets delivered during the simulation for both schemes at the two

different bit rates. As can be seen in the figure, MACMAN provides an improvement over PAC with regard to the number of data packets delivered. When using MACMAN, the flows do not need to be stopped to discover new routes. Hence more packets can be sent. At low mobility rates, the difference between MACMAN and PAC is smaller since low mobility results in fewer changes in topology. Thus there are fewer occurrences of congestion that require new routes to be found. However, as the mobility in the network increases, the difference between the protocols also increases, and MACMAN yields a delivery improvement of about 10% over PAC.

This effect can be seen even more clearly in Figure 11 where the average number of times flows are stopped per simulation is shown. When using PAC, more flows must be stopped. In many cases these same flows can switch to an alternate route and continue the transmission when MACMAN is used. There are, however, also cases where an alternate path is not present, and thus MACMAN cannot avoid flow breaks completely. Nevertheless, the decrease in the number of flow breaks confirms that alternate paths do exist in the network. This is critical since the benefit of MACMAN over PAC relies on the existence of alternate paths. As expected, the number of flow breaks increases for both protocols as the mobility in the network increases. However, MACMAN is able to handle the increased mobility better than PAC and has 20-30% fewer flow breaks than PAC. The increased number of packets that can be delivered, and, even more importantly, the reduced number of flow breaks that result from MACMAN, are likely to increase the perceived quality of service at the end users.

Since admission control protocols are primarily designed to accommodate real-time applications that have strict requirements on end-to-end delay, it is important to make sure that the extra overhead does not add additional delay that exceeds the delay bounds of the applications. The average end-to-end delay for the flows can be seen in Figure 12. From this figure, it is clear that MACMAN is able to provide the increase in data packet delivery and reduction in flow breaks without a notable increase in the end-to-end delay.

It is also important to examine the overhead produced by the protocol since MACMAN introduces additional control messages. Figure 13 shows the average number of control packets transmitted during the simulations. As expected, the graph shows a notable increase in control traffic in the network compared to PAC. As seen in Section 5.1, the actual amount of overhead is a factor of the RCQ interval. A tradeoff exists between the amount of extra overhead and the ability to quickly switch to a new path if the current one breaks. However, although there is an increase in overhead, this increase is tolerable due to the improvement of the other performance parameters. At first, it may not be intuitive as

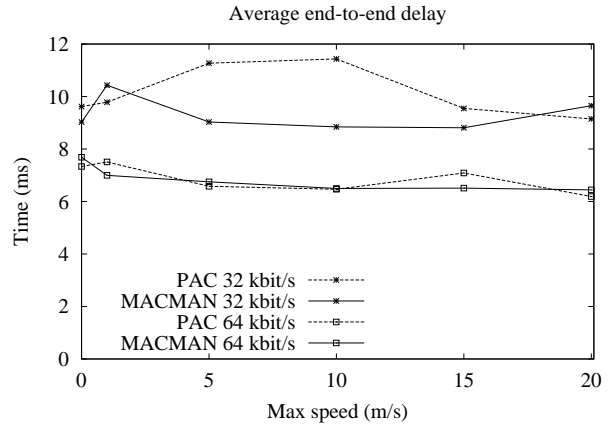


Figure 12. Average end-to-end delay

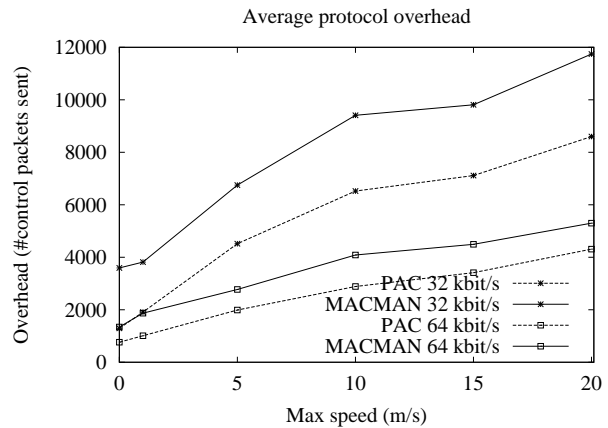


Figure 13. Average control packet overhead.

to why the overhead is higher when flows with the lower data rate are used, but it can be explained. The transmission of flows with a lower data rate allows more flows to be simultaneously admitted. Since route maintenance messages are generated on a per-flow basis for admitted flows, the overhead increases.

Finally, we examine the number of flows that each protocol allows to be simultaneously active in Figure 14. It can be seen that at lower bit rates, the difference between the protocols is negligible; however, at a bit rate of 64 kbps, MACMAN often is able to maintain one more flow than PAC. This contributes to the differences in delivery rate between the protocols and demonstrates that MACMAN is able to use the resources in the network more efficiently than PAC.

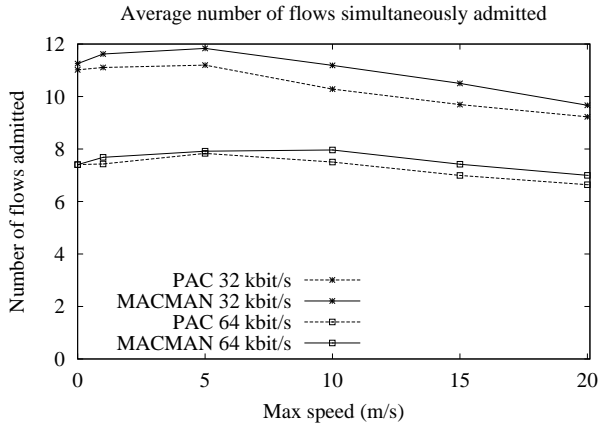


Figure 14. Average number of simultaneously admitted flows.

6. Conclusion

This paper proposes MACMAN, an improvement to existing protocols for admission control in multi-hop mobile ad hoc networks. We present a novel route maintenance scheme that discovers and maintains multiple routes to meet the quality of service needs of real-time applications. Our scheme includes a mechanism to factor out the consumption of resources by current data transmissions along alternate paths. Simulations show that our proposed solution is able to prevent disruptions in communication when mobility breaks the quality of service of an already used path. Further, we show that MACMAN increases the number of packets that are successfully delivered to destinations. At the same time, the number of times an already admitted flow has to be stopped, causing an unacceptable break in the communication, is decreased. Finally, MACMAN is able to deliver the packets with low delay, which is essential for real-time traffic. Based on its performance, we envision the utility of MACMAN as a solution for the support of multiple simultaneous data sessions with varying quality of service needs.

References

- [1] M. Barry, A. T. Campbell, and A. Veres. Distributed control algorithms for service differentiation in wireless packet networks. In *Proceedings of IEEE INFOCOM*, Barcelona, Spain, 2001.
- [2] M. Benveniste, G. Chesson, M. Hoeben, A. Singla, H. Teunissen, and M. Wentink. EDCF proposed draft text. IEEE working document 802.11-01/131r1, March 2001.
- [3] T. Camp, J. Boleng, and V. Davies. A survey of mobility models for ad hoc network research. *Wireless Commu-*

- nications & Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications*, 2(5):483–502, 2002.
- [4] E. Carlson, C. Bettstetter, H. Karl, C. Prehofer, and A. Wolisz. Distributed maintenance of resource reservation paths in multihop 802.11 networks. In *Proceedings of the IEEE Vehicular Technology Conference 2004-Fall*, Los Angeles, CA, September 2004.
- [5] I. D. Chakeres and E. M. Belding-Royer. PAC: Perceptive admission control for mobile wireless networks. In *Proceedings of the First International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks (QShine)*, Dallas, TX, October 2004.
- [6] I. D. Chakeres, J. Macker, and E. M. Belding-Royer. Perceptive admission control: A simple mote-based scenario. Poster: The Tenth Annual International Conference on Mobile Computing and Networking (Mobicom 2004), September 2004.
- [7] D. N. Hatfi eld. Technological trends in wireless telecommunications. Technical report, Gallaudet University, 1996.
- [8] D. B. Johnson and D. A. Maltz. Dynamic source routing in ad hoc wireless networks. In Imielinski and Korth, editors, *Mobile Computing*, chapter 5, pages 153–181. Kluwer Academic Publishers, 1996.
- [9] A. Lindgren, A. Almquist, and O. Schelén. Quality of service schemes for IEEE 802.11 wireless LANs - an evaluation. In *Special Issue of the Journal on Special Topics in Mobile Networking and Applications (MONET) on Performance Evaluation of QoS Architectures in Mobile Networks*, 8(3):223–235, June 2003.
- [10] C. E. Perkins, E. M. Royer, S. R. Das, and M. K. Marina. Performance comparison of two on-demand routing protocols for ad hoc networks. *IEEE Personal Communications Magazine special issue on Ad hoc Networking*, pages 16–28, February 2001.
- [11] P. Sambasivam, A. Murthy, and E. Belding-Royer. Dynamically adaptive multipath routing based on AODV. In *Proceedings of The Third Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net 2004)*, Bodrum, Turkey, June 2004.
- [12] K. Sanzgiri, I. D. Chakeres, and E. M. Belding-Royer. Determining intra-flow contention along multihop paths in wireless networks. In *Proceedings of the IEEE BroadNets 2004*, San Jose, CA, October 2004.
- [13] J. L. Sobrinho and A. S. Krishnakumar. Quality-of-Service in ad hoc carrier sense multiple access networks. *IEEE Journal on Selected Areas in Communications*, 17(8):1353–1368, August 1999.
- [14] The Institute of Electrical and Electronics Engineers, Inc. *IEEE Std 802.11 - Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications*. The Institute of Electrical and Electronics Engineers, Inc., 1999 edition.
- [15] N. H. Vaidya, P. Bahl, and S. Gupta. Distributed fair scheduling in a wireless LAN. In *Sixth Annual International Conference on Mobile Computing and Networking*, Boston, MA, August 2000.
- [16] 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.