Priority Based Multiple Access for Service Differentiation in Wireless Ad-Hoc Networks

Yu Wang[1] and Brahim Bensaou[2]

[1] Department of Electrical Engineering, National University of Singapore,
10 Kent Ridge Crescent, Singapore 119260
engp8843@nus.edu.sg
[2] Centre for Wireless Communications, National University of Singapore,
20 Science Park Road, #02-34/37, Singapore 117674
brahim@cwc.nus.edu.sg

Abstract. This article describes the Priority Based Multiple Access (PriMA) protocol, a new medium access control (MAC) protocol for single-channel ad-hoc networks. Unlike previously proposed protocols, PriMA takes into account the QoS requirements of the packets queued in stations to provide each station with a priority-based access to the channel. The direct support of PriMA for ad-hoc routing is that when some stations act as hubs in the routing structure and route packets for other stations besides their own, they can have high priorities and obtain larger share of bandwidth. Simulation results show the potential benefits that PriMA brings about to ad-hoc networks, and confirm PriMA as an initial step towards QoS provision in ad-hoc networks.

1 Introduction

An ad-hoc network is a dynamic multi-hop wireless network that is established by a group of mobile stations without the aid of any pre-existing network infrastructure or centralized administration. It can be installed quickly in emergency or some other special situations and is self-configurable, which makes it very attractive in many applications. Some applications include remote sensing (e.g. earthquakes, fire, environmental data gathering), robotic communication (e.g. cleaning, firefighting, patrolling, oceanic fishing, farming), Internet access and LEO constellations. Internet Engineering Task Force (IETF) has set up a workgroup named Mobile Ad-hoc Networks (MANET) [1] to develop and evolve MANET routing specifications and introduce them to the Internet Standards track. Recent discussions in MANET's mailing list on MANET's application scenarios can give an idea of the fields when MANETs can be deployed (e.g. an ad-hoc network of New York taxi cabs).

An efficient medium access control (MAC) protocol through which mobile stations can share a common broadcast channel is essential in an ad-hoc network because the medium or channel is a scarce resource. Due to the limited transmission range of mobile stations, multiple transmitters within range of the

same receiver may not know one another's transmissions, and thus in effect "hidden" from one another. When these transmitters transmit to the same receiver at around the same time, they do not realize that their transmissions collide at the receiver. This is the so-called "hidden terminal" problem [2] which is known to degrade throughput significantly. Due to their multi-hop characteristics, ad-hoc networks suffer much more from the hidden terminal problem than wireless LANs do.

To address the hidden terminal problem, various distributed MAC protocols were proposed in the literature. MACA (Multiple Access Collision Avoidance) protocol [3] proposed exchange of short Request-to-Send (RTS) and Clear-to-Send (CTS) packets between a pair of sender and receiver before actual data packet transmission and formed the basis for several other more sophisticated schemes. Among them, FAMA-NCS (Floor Acquisition Multiple Access with Non-persistent Carrier Sensing) [4] is immune to the hidden terminal problem and can achieve good throughput in ad-hoc networks. IEEE 802.11 committee also proposed a MAC protocol called Distributed Foundation Wireless Medium Access Control (DFWMAC) for wireless ad-hoc LANs [5], which in essence is a variant of CSMA/CA protocols. DFWMAC provides basic and RTS/CTS access method. Chhaya and Gupta [6] have shown that the performance of RTS/CTS access method degrades much slower than the basic access method when the number of hidden terminals is large, or the offered load is significantly larger than the channel capacity; therefore, the RTS/CTS method is more robust to fluctuations in parameter values which are common in ad-hoc networks. In DFW-MAC RTS/CTS access method, 4-way dialog including RTS-CTS-DATA-ACK is used to combat the hidden terminal problem. DFWMAC, however, still cannot prevent data packets from colliding with control packets (RTSs and CTSs) and other data packets, instead, it uses a sophisticated modified binary exponential backoff scheme to quickly resolve collisions and thus increase throughput.

These MAC protocols perform well in solving hidden terminal problem, however, none of them takes any step towards providing packet level QoS parameters such as packet loss ratio, packet delay, etc. They can support only best effort delivery service, thus limit the applications of ad-hoc networks. Recently, there has been a surge in modifying and extending equality-based medium access schemes to support priority-based access when the packets queued at stations have different QoS requirements. Although this research effort does not compete with the effort deployed in the Wireless LAN (infrastructure based), some researchers have been working on QoS support for distributed wireless networks [7, 8]. In [7], stations with real-time packets in the queue would jam the channel with Black Bursts (BBs) whose length is proportional to the delay incurred. The station that sends the longest BBs wins access to the channel and can transmit its packet thereafter. However, this approach fails when hidden terminals exist as those hidden terminals may have experienced the same delay and each BB contention period is not guaranteed to produce a unique winner, thus real-time data packets will still suffer from collisions. In [8], GAMA (Group Allocation Multiple Access) protocol was proposed for scheduling real-time and datagram traffic in

a single-hop wireless ad-hoc LANs. GAMA includes contention period during which stations can send request to join transmission group and contention-free period during which stations in transmission group take turns to transmit packets. This approach does not work well if hidden terminal exists. If hidden terminals do not join the transmission group that it may interfere with, GAMA cannot ensure data packets to be free from collision. If hidden terminals do join the transmission group to avoid collision, following the same logic, then all the other stations in the network have to join the same transmission group one by one. It is very difficult to maintain the global group due to the dynamic nature of ad-hoc network. In addition, we cannot benefit from spatial reuse. Therefore, these protocols that were originally proposed for wireless LANs are not directly applicable to multi-hop ad-hoc networks. We believe that a suitable MAC protocol for ad-hoc networks should address both hidden terminal problem and QoS issues at the same time.

Another motivation for priority-based access is to provide better support for ad-hoc routing. On one hand, those protocols originally proposed for wireless LANs normally do not take routing into account as they expect a wireless access point that can reach all other stations and relay packets for them to be deployed. Therefore they are more fit for networks with infrastructure than ad-hoc networks where routing is another major issue. On the other hand, unlike in conventional wired networks, any host that act as a router in ad-hoc networks normally has only one network interface, there are no separate links for them to route packets or exchange routing information. Especially when some stations in an ad-hoc network behave as cluster heads [9] or belong to the core of the routing structure [10], more traffic will transit through them. Obviously, they need higher priorities in accessing the channel to route packets for others in addition to their own.

In this paper, we propose a new protocol named Priority Based Multiple Access (PriMA) to support differentiated access priorities to the channel. It implements MAC-level acknowledgment as in DFWMAC while adopting the collision-free data transmission characteristics of FAMA-NCS. More importantly, it implements a novel distributed scheduling algorithm which gives stations a dynamic-priority based access to the channel by taking into account both the packet delay requirement and the packet loss ratio incurred by the ongoing session. The remainder of the paper is organized as follows. Section 2 describes PriMA protocol in detail. Section 3 compares by simulation the performance of FAMA-NCS, IEEE 802.11 DFWMAC and PriMA. Section 4 concludes this paper.

2   PriMA Protocol

2.1   Overview

PriMA protocol requires a station that wishes to send a data packet to acquire the channel before transmitting the packet. The channel is acquired by establishing an RTS-CTS dialog between the sender and the receiver. Although multiple

control packets may collide, data packets are always sent free from collisions. This is achieved by the enforced requirements of size relationship between RTS and CTS as well as different periods a station should wait after receiving a packet or sensing the channel busy. Fullmer et al. [11] have given detailed description about this. They showed that as long as the length of CTS packet is sufficiently longer than that of the RTS packet, CTS can act as a jamming signal to prevent other stations from transmission. Stations that hear the channel busy should wait long enough for the possible ongoing data transmission to go on unobstructed. PriMA's trivial extension to this is to reply an acknowledgment packet after successful reception of a data packet. This is also the common case in some other MAC protocols like DFWMAC. The extension of MAC-level acknowledgment still ensures data transmission free from collisions.

The most important feature of PriMA is that the access to the channel is based on priorities of the packets queued in stations. This is achieved by three timers calculated according to the QoS requirements of the data packets in each station. The first is the *access timer* that a station should wait after the channel becomes idle before transmitting an RTS packet. The access time is similar to DIFS (Distributed InterFrame Space) in DFWMAC and the silent period of channel in FAMA-NCS. However, unlike the fixed length of DIFS and the random length of the silent period in FAMA-NCS, the access time is dynamically adjusted based on the QoS requirements. The second is the *delay timer* carried in every RTS packet. It indicates how long the intended receiver can wait before replying with a CTS packet. Therefore, an earlier sent RTS packet may be preempted by a later sent RTS packet which indicates a higher priority data packet transmission request. The third is the *backoff time* that a station should wait before retransmission when collision occurs. The backoff time is uniformly distributed, however the upper bounds of the distribution vary among stations. Stations that hold higher priority packets have lower upper bounds. Therefore, they can statistically recover earlier than other stations and bid for the channel again. For stations queued with normal data packets, we just set a large upper bound comparable to that of DFWMAC. Therefore, our backoff scheme is not necessarily more prone to collapse than binary exponential backoff with upper bound.

2.2   Description of PriMA

To simplify our description of PriMA, the processing time and transmit-to-receive turnaround time are ignored. We say a station "detects" collisions if it senses the channel busy without being able to receive any intelligible packet. Following, we define some of the notations used in this section:

$rtPacket$: Data packet with QoS parameters specified
$nrtPacket$: Data packet without any QoS parameters specified
T_d : Maximum one-hop channel propagation delay

T_{type} : Time to transmit a packet of type *type* where *type* can be either RTS, CTS, DATA or ACK

T_{delay} : Allowed delay time to reply to RTS, carried in RTS packet

T_{left} : Time left for a *rtPacket* to be delivered; when T_{left} is less than a threshold, the corresponding *rtPacket* will be dropped.

T_{max} : Maximum time to complete one successful RTS-CTS-DATA-ACK transmission

T_{access} : Time required to sense the channel idle before transmitting an RTS packet

T_{defer} : The longest time a station should defer access when "detecting" collisions before entering backoff. It equals $T_{data} + 3 * T_d$.

T_{unit} : Time used as a factor to map packet delay requirement to backoff timer

N_{lost} : Number of packets dropped during the session

N_{sent} : Number of packets sent during a session

PLR : Packet Loss Ratio (QoS requirement).
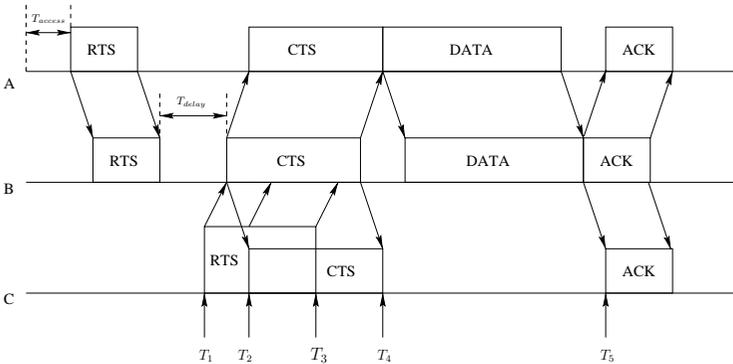
Fig. 1. PriMA illustration

Figure 1 shows how PriMA operates. In this figure, stations A and C are hidden from each other and station B is a neighbor of both A and C. As in all other collision avoidance multiple access techniques PriMA uses the RTS-CTS combination to implement collision avoidance, besides to ensure the problem of hidden terminals is addressed adequately, PriMA imposes FAMA's conditions on the sizes of the RTS and CTS packet [11]:

$$- \ T_{rts} > 2 * T_d$$
$$- \ T_{cts} > T_{rts} + 2 * T_d$$

and stations that sense the channel busy should wait for T_{defer} to let the possible ongoing data transmission to finish unobstructed. FAMA's rational in imposing these conditions, is that although C is hidden from A, and thus C did not hear

A's RTS, if C sends an RTS packet that at most collides with the CTS reply from B to A, the condition ensures that C hears the trailer of the CTS and thus C would abort any transmission and enters the backoff procedure. This ensures that the data packet from A to B is transmitted collision free. Note that it is not necessary to adopt these restrictions (enforced in FAMA) in PriMA to provide soft QoS. Nevertheless, we believe that it does not make sense to address the problem of QoS (be it soft QoS) without addressing first the hidden terminal problem, especially when the offered traffic load can be very high. PriMA's approach to providing QoS can also be applied to IEEE 802.11's DFWMAC.

In PriMA, any station that has a packet to send should wait until it senses the channel idle for a certain amount of time called *access timer* before transmitting an RTS request. The access time calculation, shown in Algorithm 1, is based on the required QoS and the perceived QoS.

Algorithm 1 Access timer calculation

1: **if** (nrtPacket or rtPacket with $T_{left} > 1$ sec) **then**
2:    $T_{access} = 2 * T_d$
3: **else**
4:    **if** ($T_{left} > 0$) **then**
5:       $T_{access} = T_{left} - (N_{lost} - PLR * N_{sent}) * T_{max}$
6:       **if** ($T_{access} < 0$) **then**
7:          $T_{access} = 0$
8:       **else**
9:          $T_{access} = 2 * T_d / |\log_2 (T_{access}/2)|$
10:       **end if**
11:    **end if**
12: **end if**

Note that in order to reduce the waste of bandwidth due to this access time, time-sensitive packets which have a time to live of more than one second, compete at the same priority level as the non time sensitive packets. These packets would gain higher priorities when they become more urgent. Note that initially, at the packet's first attempt, T_{left} is initialized to a value proportional to the maximum tolerable delay of the packet. Also, throughout the algorithm, T_{left} decreases with the ticks of the clock in the same proportion. The calculation in line 5 of the algorithm shows that the more has a station suffered excess droppped packets ($N_{lost} - PLR * N_{sent}$), the shorter T_{access} will be. This excerpt of pseudo-code clearly shows that when two stations attempt to access the channel simultaneously T_{access} will differentiate their attempts according to their packets' QoS requirement.

When a station succeeds in its access to the channel, it sets a "delay" field in its RTS packet and sends it out. The value of the field is denoted by T_{delay}, which indicates how long the intended receiver can wait before replying with a CTS packet. T_{delay} is calculated as shown in Algorithm 2.
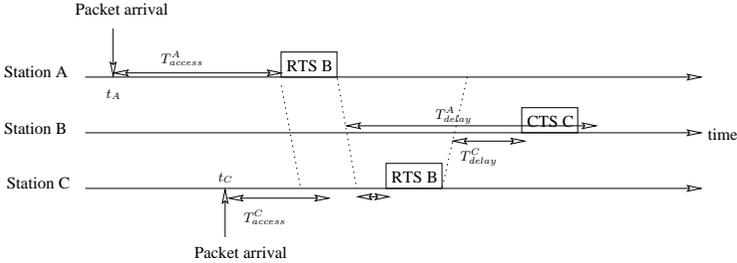

Algorithm 2 Delay timer calculation

1: **if** (nrtPacket) **then**
2:     $T_{delay} = 2 * T_{rts}$
3: **else**
4:     **if** $(T_{left}/T_{max} > 2)$ **then**
5:         $T_{delay} = 2 * T_{rts}$
6:     **else**
7:         $T_{delay} = (T_{left}/T_{max}) * T_{rts}$
8:     **end if**
9: **end if**

Fig. 2. Illustration of the preemption in PriMA

To illustrate the importance of this second timer, let us consider the example in Figure 2 where two stations, A and C, attempt to communicate with station B. We will neglect the propagation times in this example. C's data packet arrives at the MAC layer at time t_C later than A's data packet who arrives at t_A, however, C's delay target is more stringent than A's. Due to the lack of a global coordinator in the system and the lack of a global state, there is no way for C to succeed in sending an RTS packet to B before A does, since $t_C + T_{accces}^C > t_A + T_{access}^A$. A sets a value T_{delay}^A in its RTS packet which tells B the amount of time it can wait before replying with a CTS. After C hears the end of A's RTS (we will see later the case where A and C are hidden from each other), it will apply the same procedure it uses for the first access, however with only the remaining access timer. In other words, C will wait until it senses an idle channel for $T_{left} = t_C + T_{accces}^C - t_A - T_{access}^A$ starting at $t_A + T_{access}^A + T_{RTS}$. If this left time is smaller than T_{delay}^A, i.e., if C's access timer expires before B replies with a CTS to A than C can send to B an RTS that will preempt A's RTS. This way, although A succeeded in sending the RTS before C, this latter has the possibility to preempt A before it starts transmitting its data packet.

In the calculation of T_{delay} for a real time packet, only T_{left} is used. This is because T_{left} is a composite value affected by both packet delay and packet drop ratio. Thus we can keep the algorithm simple yet effective. This algorithm shows that the RTS packet for a data packet that is more delay sensitive will have shorter T_{delay} when the data packet is about to expire. The calculation will permit the packets to gain higher priorities after they are delayed for some

time. Stations usually do not reply with CTS immediately after receiving an RTS packet unless required to do so, therefore a station may receive multiple RTS packets in a row. If the later arrived RTS packet requires shorter delay than that of the earlier one, the station can reply to it first. This makes it possible for delay sensitive data packets to preempt other data packets. In addition, T_{delay} does not exceed two times the RTS packet transmission time to minimize the protocol overhead.

It is inevitable that control packets may collide with each other, therefore those stations that detect collisions should also wait for T_{defer} and then back off for a random time $T_{backoff}$, which is calculated as shown in Algorithm 3, where $U(0, x)$ is a uniformly distributed random number in the interval $(0, x)$.

Algorithm 3 Backoff timer calculation

1: **if** (nrtPacket or rtPacket with $T_{left}/T_{unit} > Maxtimer$) **then**
2:     $T_{backoff} = U(0, Maxtimer) * 2T_d$
3: **else**
4: $T_{backoff} = U(0, T_{left}/T_{unit}) * 2T_d$
5: **end if**

The value of *Maxtimer* is simply set to a large value, say 800, which is comparable to the maximum timer used in DFWMAC. For example, the IEEE 802.11 Wireless LAN standard specification for direct sequence spread spectrum (DSSS) uses a timer (or *Contention Window*, in 802.11's terminology) ranging from 31 to 1023. In PriMA, $T_{backoff}$ is based on a uniform distribution whose upper bound value varies according to the delay requirement of a data packet. This calculation will statistically give stations that have delay sensitive data packets shorter $T_{backoff}$. Thus, on average, these stations will end the backoff period earlier than other stations and bid for the channel again. To illustrate how Algorithm 3 contributes in differentiating the stations, let us return to the example of Figure 2. In the case where A and C are hidden from each other. C cannot hear the RTS packet of A and thus after exhausting its timer T_{access}^C, it would also send an RTS packet which would collide at B with A's RTS packet. After T_{delay}^A (respectively T_{delay}^C) station A (respectively station C) notices the collision by timeout, and thus they both apply the above backoff algorithm with their respective QoS targets. In this case, C has higher probability in bidding for the channel before A does, since C draws its uniformly distributed backoff time from a smaller interval than A does, and thus C has higher probability to terminate its backoff before A.

From the above descriptions, we can see that by dynamically adjusting three timers, T_{access}, T_{delay} and $T_{backoff}$, according to the data packets' QoS requirements, PriMA can give stations priority-based access to the channel. At the same time, various waiting time periods are carefully defined to prevent control packets from colliding with data packets.

3   Simulation Results

PriMA is a complex MAC protocol for analytical modelling because of the danymic nature of the different timers, thus in this paper the performance of PriMA is investigated by simulations. In our experiments, we investigate symmetrical networks where each station has N neighbors and is hidden from Q neighbors of any one of its neighbors, thus each station has the same spatial characteristics. Figure 3 shows two sample configurations for $N = 4$, $Q = 2$ and $Q = 3$. For example, for station A in figure 3(a), it has 4 neighbors including station B and is hidden from station B's 3 neighbors: C, D and E. As this graph can grow to infinitely large, we fold or collapse it so that the total number of stations in the network is limited while symmetry is maintained. The resulting graphs are shown in figure 4.

(a) N=4, Q=3   (b) N=4, Q=2
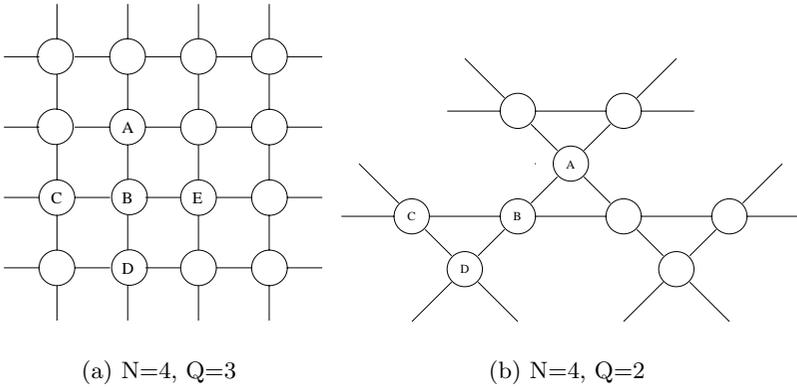
Fig. 3. Two sample configurations

(a) N=4, Q=3   (b) N=4, Q=2
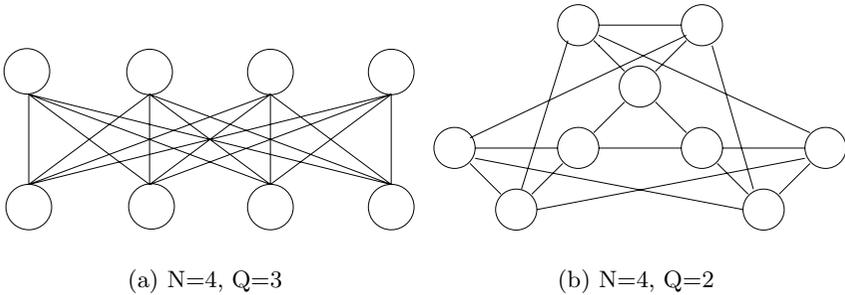
Fig. 4. Two sample configurations (collapsed)

We assume a 1Mbps ideal channel with zero preamble and processing overhead. We have performed different sets of simulations with OPNET Modeler/Radio and we compare our results to those provided by alternative protocols, notably FAMA-NCS and IEEE 802.11 DFWMAC[1]. Table 1 lists the parameters used in the simulation. As we ignore the extra time incurred by hardware and software, the different InterFrame Spaces (IFSs) in IEEE 802.11 are reduced accordingly and they are shown in table 2.

Protocol	RTS	CTS	DATA	ACK	backoff timer	backoff unit time
FAMA	20-byte	25-byte	500-byte	-	10	$160\mu sec$
IEEE 802.11	25-byte	20-byte	500-byte	20-byte	31-1023	$6\mu sec$
PriMA	20-byte	25-byte	500-byte	20-byte	800	$12\mu sec$

Table 1. Protocol configuration parameters

DIFS	SIFS	EIFS
$12\mu sec$	$0\mu sec$	1.3msec

Table 2. Extra configuration parameters for IEEE 802.11

In the first set of simulations, all stations generate Poisson traffic with the same mean rate and all require best-effort delivery service. The simulation measures the throughput of the protocol against the degree of the nodes. The results are shown in Figure 5. For comparative purpose, we also show the analytical results for Slotted-ALOHA with separate acknowledgement channel and Non-Persistent CSMA in these figures. The figures demonstrate that FAMA's performance degrades dramatically when the number of competing stations (including neighbors and hidden terminals) increases. This is due to the ineffectiveness of uniform backoff scheme with small timer in collision resolution used in FAMA. However, it still performs well in other situations because of its immunity to hidden terminal problem. IEEE 802.11 performs quite well in all the situations because of its modified binary exponential backoff scheme. Although PriMA's performance is not quite outstanding when the number of competing stations is small, however, PriMA does achieve a rather stable throughput and is comparable to IEEE 802.11 when the number of competing stations increases. This is not surprising. PriMA, by introducing the different timers as well as the MAC acknowledgments in fact sacrifices throughput to achieve different priorities among the stations (which we show later). In addition, large backoff timers will become more effective when the number of competing stations increases, therefore the throughput difference between PriMA and IEEE 802.11 is decreasing. In other

---

[1] We use its specification for Direct Sequence Spread Spectrum if applicable.

words, PriMA captures in fact the effectiveness of FAMA in solving the hidden terminal problem as well as 802.11's ability in sustaining throughput at high loads. It sacrifices however a little throughput by introducing the different timers in order to achieve differentiation between the stations as shown in the following.

In the second set of simulations, each station still generates Poisson traffic with the same mean rate. However, we set one station (router) to generate packets that require delay-bounded delivery, while the other stations (hosts) have no delay requirement. This, in essence, makes the packets from the router have higher priority than packets from the hosts.

This scenario may be applicable in a case when some stations form a group and choose one of them to act as a router. In other words, the router will handle a larger amount of traffic and thus needs some priority to access the channel. Simulation results are shown in figure 6 and 7 with different values of N and Q. The figure clearly shows that the router has higher throughput than other stations despite the fact that they all offer the same traffic load to the channel. Even when the number of competing stations increases, the router can still achieve much higher throughput than other normal stations. This differentiation cannot be achieved by FAMA-NCS and other non prioritized protocols such as IEEE 802.11 DFWMAC.

In the third set of simulations, we select two stations (high priority router and low priority router) to generate constant rate packets that require different packet delivery delay bounds. We investigate two performance metrics under different scenarios: namely, the packet loss ratio (due to expiration), and the average delay. To get comparative results, we set the packet interarrival times equal to the packet delay requirement, otherwise protocols like IEEE 802.11 will never keep up with the packet generation rate and eventually will drop nearly all the packets. Simulation results are shown in figure 8. The first three sub-figures (8(a), 8(b), 8(c)) show the packet loss ratio against the offered load. They show that when the offered load increases, the packet drop ratio for IEEE 802.11 increases significantly while PriMA can still maintain low packet drop ratio for the two routers. Figure 8(d) shows that the average packet delay achieved by the two types of routers in PriMA. The figure clearly shows that when the traffic load is very low, the packets from all the stations experience the same (insignificant) delay. However, when the traffic load increases, the protocol starts differentiating the stations according to their requirements.

4   Conclusion

PriMA is a new MAC protocol that is specifically designed for ad-hoc networks. It can achieve good throughput in ad-hoc networks where hidden terminal problem is common. In addition, every station has priority-based access to the channel, thus PriMA can provide elementary QoS support from the bottom up, making it a good choice for supporting higher layer protocols that require quality of service. Another benefit of PriMA is that it can provide better support for
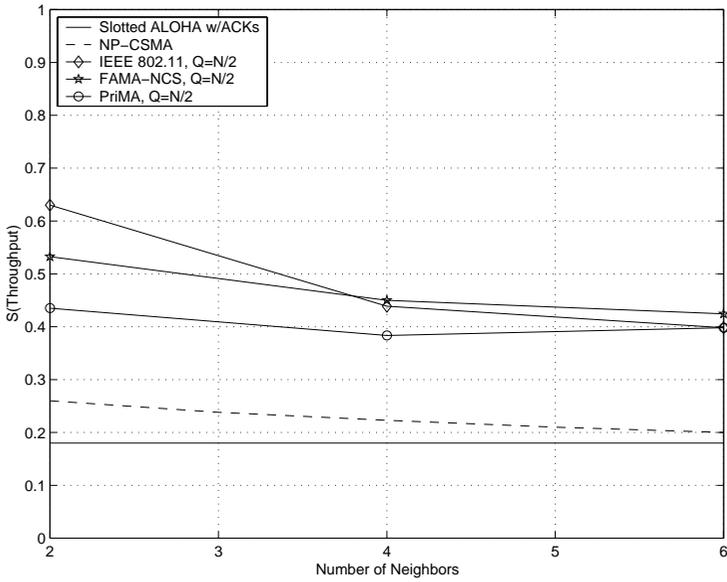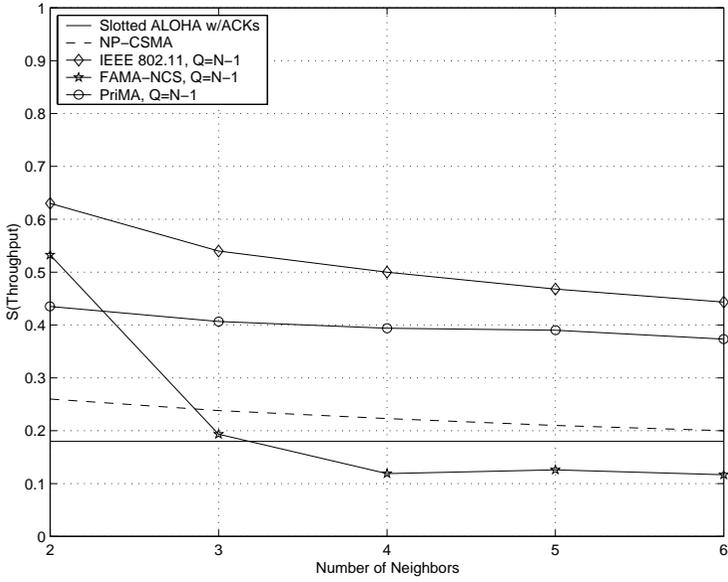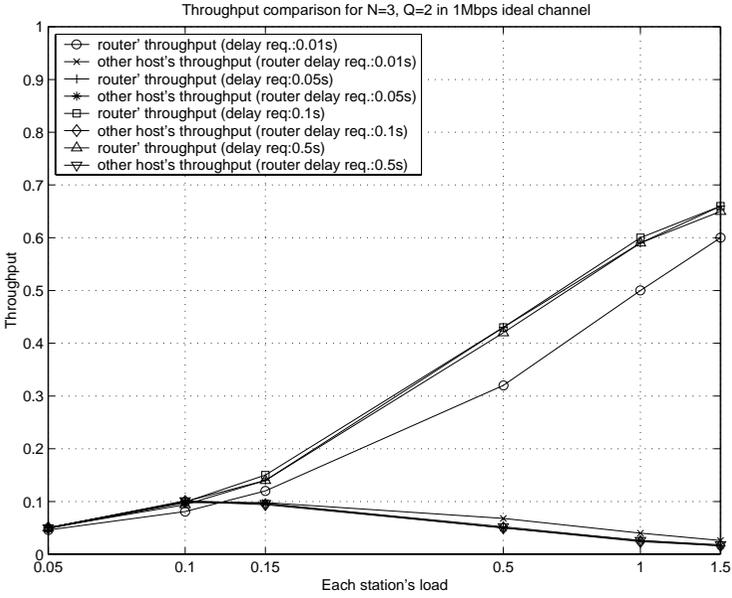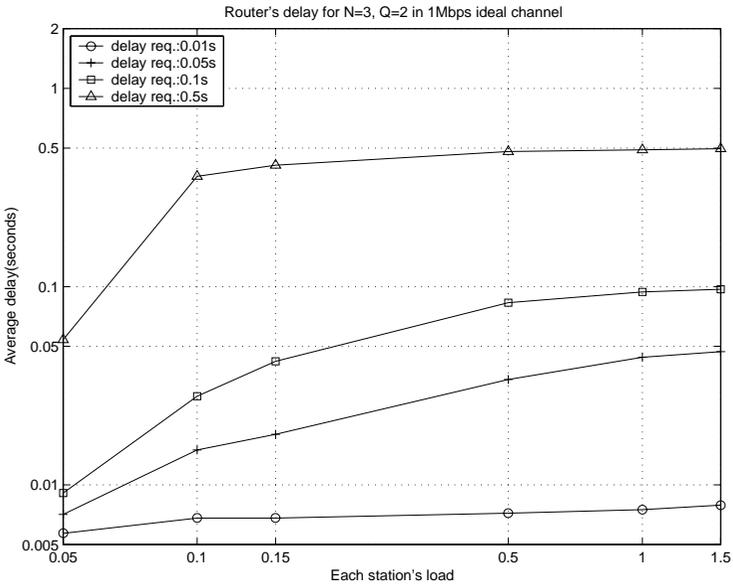
Fig. 5. Throughput versus node degree

(a)

(b)

Fig. 6. One router configuration (N=3)

Throughput comparison for N=4, Q=3 in 1Mbps ideal channel

(a)

Router's delay for N=4, Q=3 in 1Mbps ideal channel
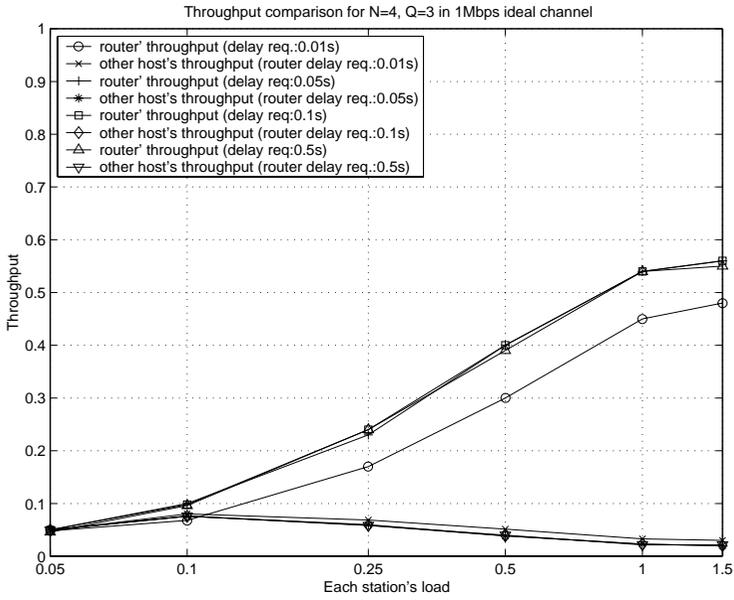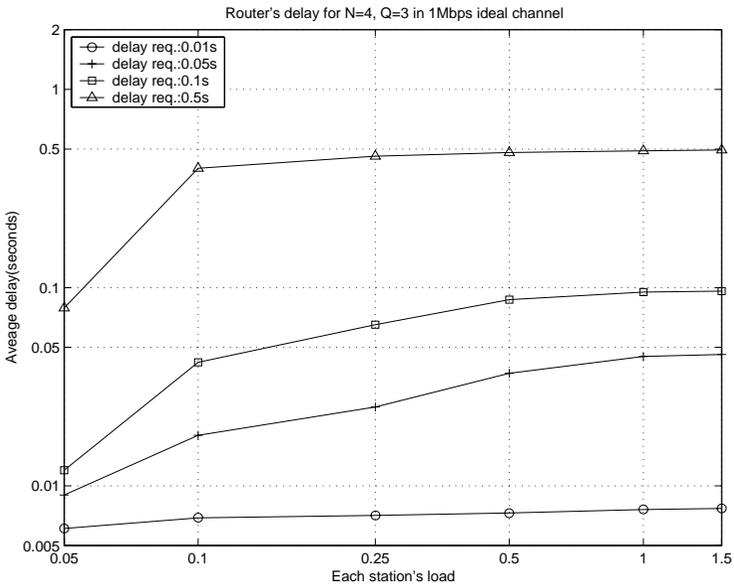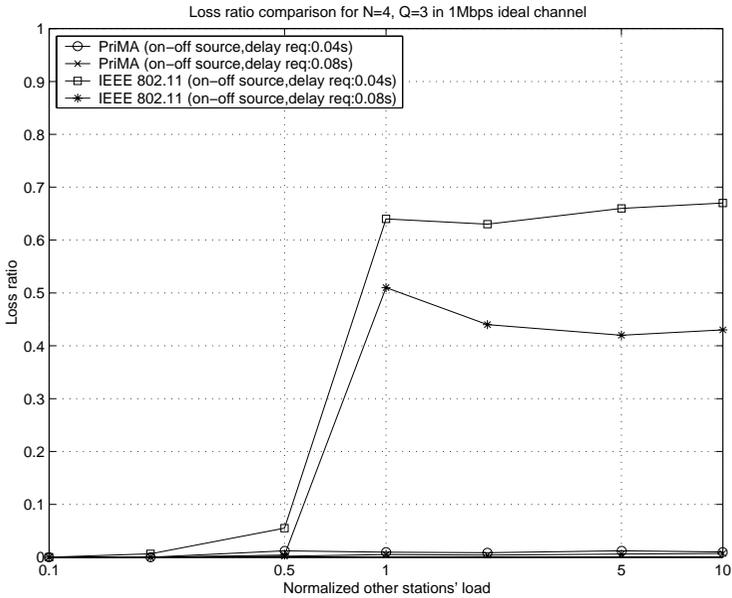

(b)

Fig. 7. One router configuration (N=4)

Loss ratio comparison for N=4, Q=3 in 1Mbps ideal channel

- PriMA (on−off source,delay req:0.03s)
- PriMA (on−off source,delay req:0.06s)
- IEEE 802.11 (on−off source,delay req:0.03s)
- IEEE 802.11 (on−off source,delay req:0.06s)

Loss ratio

Normalized other stations' load

(a)

Loss ratio comparison for N=4, Q=3 in 1Mbps ideal channel

- PriMA (on−off source,delay req:0.04s)
- PriMA (on−off source,delay req:0.08s)
- IEEE 802.11 (on−off source,delay req:0.04s)
- IEEE 802.11 (on−off source,delay req:0.08s)
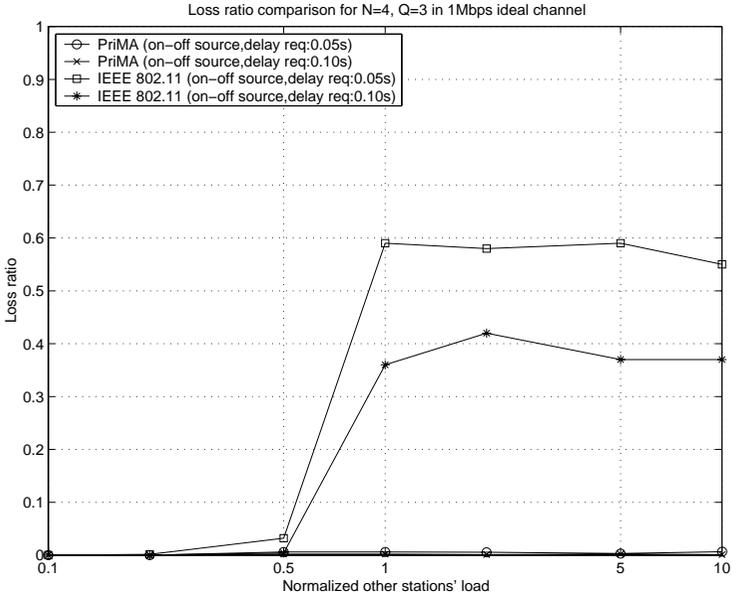
Loss ratio

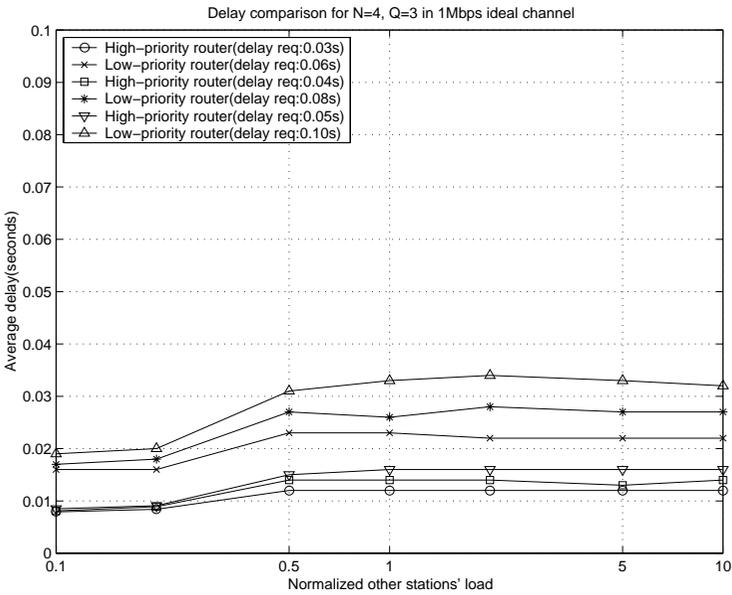Normalized other stations' load

(b)

Fig. 8. QoS differentiation

(c)

(d)

Fig. 8. QoS differentiation (con't)

ad-hoc routing as some packets can be given higher priorities and get delivered earlier than others. Simulation results show that PriMA is an especially promising MAC protocol for ad-hoc networks in presence of heterogeneous traffic and QoS requirements.

References

[1] http://www.ietf.org/html.charters/manet-charter.html.

[2] F. A. Tobagi and L. Kleinrock, "Packet Switching in Radio Channels: Part II - the Hidden Terminal Problem in Carrier Sense Multiple-access Modes and the Busy-tone Solution," in *IEEE Transactions on Communications, vol. COM-23, no. 12*, pp. 1417–1433, 1975.

[3] P. Karn, "MACA - a New Channel Access Method for Packet Radio," in *ARRL/CRRL Amateur Radio 9th Computer Networking Conference*, pp. 134–140, ARRL, 1990.

[4] J. Garcia-Luna-Aceves and C. L. Fullmer, "Performance of Floor Acquisition Multiple Access in Ad-Hoc Networks," in *Proceedings of 3rd IEEE ISCC*, 1998.

[5] IEEE Computer Society LAN MAN Standards Committee, ed., *IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. IEEE Std 802.11-1997, The Institute of Electrical and Electronics Engineers, New York, 1997.

[6] H. S. Chhaya and S. Gupta, "Throughput and Fairness Properties of Asynchronous Data Transfer Methods in the IEEE 802.11 MAC Protocol," in *6th International Conference on Personal, Indoor and Mobile Radio Communications*, 1995.

[7] J. L. Sobrinho and A. S. Krishnakumar, "Quality-of-Service in Ad Hoc Carrier Sense Multiple Access Wireless Networks," in *IEEE Journal on Selected Areas in Communications, Vol. 17, No. 8*, pp. 1353–1368, 1999.

[8] A. Muir and J. J. Garcia-Luna-Aceves, "Supporting real-time multimedia traffic in a wireless LAN," in *Proc. SPIE Multimedia Computing and Networking*, pp. 41–54, 1997.

[9] M. L. Jiang, J. Y. Li, and Y. C. Tay, "Cluster Based Routing Protocol(CBRP) Functional Specification." draft-ietf-manet-cbrp-spec-00.txt, Aug. 1998. Work in progress.

[10] R. Sivakumar, P. Sinha, and V. Bharghavan, "Core Extraction Distributed Ad hoc Routing (CEDAR) Specification." draft-ietf-manet-cedar-spec-00.txt, Oct. 1998. Work in progress.

[11] C. L. Fullmer and J. J. Garcia-Luna-Aceves, "Solutions to Hidden Terminal Problems in Wireless Networks," in *Proceedings of ACM SIGCOMM*, 1997.