Poll-before-Data Multiple Access

Asimakis Tzamaloukas and J.J. Garcia-Luna-Aceves
Computer Engineering Department
Baskin School of Engineering
University of California, Santa Cruz, California 95064
{jamal, jj}@cse.ucsc.edu

Abstract— **We introduce a new medium access control (MAC) protocol named PDMA (Poll-before-Data Multiple Access). Most prior MAC protocols aimed at avoiding collisions of data packets in networks with hidden terminals are sender-initiated, in that the sender transmits a short request to send (RTS) asking the receiver for permission to transmit. In contrast, in PDMA, when a data packet arrives at the receiver, the receiver sends a ready-to-receive-and-transmit (RT2) packet stating the identifiers of a specific sender and receiver. A node receiving an RT2 packet addressed to it as a sender is enabled to send a data packet, if it has one; otherwise, if the sender specified in the RT2 is quiet, the receiver specified in the RT2 sends a clear-to-send (CTS) packet, enabling the sender of the RT2 to send its own data packet free of collisions. PDMA is shown to avoid the collision of data packets with any type of packet. An analytical model is used to show that PDMA achieves higher throughput than prior collision-avoidance protocols for wireless networks, namely MACA-BI, FAMA-NCS, and MACA. Average delays in PDMA are also shown to be shorter than those in CSMA.**

I. Introduction

There has been a significant amount of research on designing efficient MAC (medium access control) protocols for wireless networks. The first attempt was CSMA (carrier sense multiple access), which is based on sensing the channel before attempting to transmit a packet. Kleinrock and Tobagi [7] studied CSMA's behavior and identified the *hidden-terminal* problem of CSMA, which makes CSMA perform as poorly as the ALOHA protocol when the senders of packets cannot hear one another and the vulnerability period of packets becomes twice a packet length. The BTMA (busy tone multiple access) protocol was a first attempt to solve the hidden terminal problem by introducing a separate busy tone channel [11]. The same authors proposed SRMA [12], which attempts to avoid collisions by introducing a collision-avoidance handshake between the sender and the receiver. A node with a data packet to send sends first a request-to-send (RTS) packet to the intended receiver, who responds with a clear-to-send (CTS) if it receives the RTS correctly. The sender transmits the data packet after receiving the RTS successfully. ALOHA or CSMA can be used by the senders to transmit RTSs.

Several variations of the basic SRMA scheme to avoid collisions of data packets have been developed; MACA [6], MACAW [1], IEEE 802.11 [5], and FAMA [2], [4] are just a few examples. All of these MAC protocols, and most protocols based on collision-avoidance exchanges to date, are *sender-initiated*, in that the node wanting to send a data packet first transmits a short control packet asking permission from the receiver. In contrast, in the MACA by invitation (MACA-BI) protocol [9], [10], the collision-avoidance handshake between sender and receiver is reversed and is made *receiver initiated*. In MACA-BI, a node with a packet to send must wait for the intended receiver to send a polling packet, called RTR (ready to receive), addressed to the node, before it can transmit anything. This is an interesting approach; however, as we show in Section IV-B, it renders a low throughput, unless the average rate of packets is high, because many RTRs go unanswered.

This work was supported by the Defense Advanced Research Projects Agency (DARPA) under Grant No. F30602-97-2-0338 and by a grant from Raytheon.

We present a new receiver-initiated MAC protocol that eliminates the performance limitations of MACA-BI, which we call the Poll-before-Data Multiple Access (PDMA) protocol. PDMA turns the RTT used in MACA-BI into a dual-use control packet called RT2 (Ready To Receive and Transmit). When a node has data to send to an intended receiver, it sends an RT2 using carrier sensing, which we assume to be non-persistent in this paper. The RT2 specifies two neighbors: (a) an intended source of data, which is chosen using any polling scheme (e.g., round robin), and (b) an intended receiver of the data the node has for transmission. The RT2 leads to a successful transmission of data if either the polled source has data for the sender of the RT2, or the intended receiver sends a clear-to-send (CTS) to the node if the polled source remains quiet. Section II describes PDMA in detail, and Section III shows that PDMA correctly avoids collisions of data packets with any other packets in the absence of transmission errors and fading. Section IV shows the improvement in throughput over MACA-BI and even FAMA-NCS resulting from the modified collision-avoidance dialogue of PDMA using an analytical model.

II. PDMA

A critical design issue in any receiver-initiated MAC protocol is deciding on the frequency with which a receiver must poll its neighbors for packets. A polling rate that is too small renders low throughput and long average delays, because each sender with a packet to send is slowed down by the polling rate of the receiver. Conversely, a polling rate that is too high also renders poor performance, because the polling packets are more likely to collide with each other and no source gets polled. The basic solution advocated in PDMA consists of two elements: (a) making the data rate at the sources determine the polling rate at the receivers, and (b) eliminating wasted polling packets. To achieve the desired clocking effect of data packets over polling packets, a node sends an RT2 (ready to receive and transmit) packet when it has data to send itself. To avoid wasteful polling packets, the RT2 serves two purposes: it polls a specific neighbor asking for packets, and it asks permission from another specific neighbor to send packets if the polled source remains quiet.

In PDMA, every node starts operation in the START state, in which the node waits twice the maximum channel propagation delay, plus the hardware transmit-to-receive transition time (ϵ), before sending anything over the channel. This enables the node to find out if there are any ongoing transmissions. After a node is properly initialized, it transitions to the PASSIVE state. In all states, before transmitting anything to the channel, a node must listen to the channel for a period of time that is sufficient for the node to start receiving packets in transit. If a node x is in the PASSIVE state and senses carrier, it transitions to the REMOTE state to defer to ongoing transmissions. A node in REMOTE state must allow enough time for a complete successful handshake to take place, before attempting to transition from remote state.

Figure 1 illustrates the basic handshakes that occur in PDMA. If node x is in PASSIVE state and obtains an outgoing packet to send

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **1999**	2. REPORT TYPE	3. DATES COVERED **00-00-1999 to 00-00-1999**
4. TITLE AND SUBTITLE **Poll-before-Data Multiple Access**		5a. CONTRACT NUMBER
		5b. GRANT NUMBER
		5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)		5d. PROJECT NUMBER
		5e. TASK NUMBER
		5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **University of California at Santa Cruz,Department of Computer Engineering,Santa Cruz,CA,95064**		8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR'S ACRONYM(S)
		11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**		**5**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

to neighbor y, it transitions to the RT2 state. In the RT2 state, node x uses non-persistent carrier sensing to transmit an RT2. If node x detects carrier when it attempts to send the RT2, it transitions to the BACKOFF state, which makes the node back off immediately for a sufficient amount of time to allow a complete handshake between a sender-receiver pair to occur; otherwise, x sends its RT2.

The RT2 specifies two addresses: the address of a neighbor z, which is the intended source being polled and can be chosen using any neighbor polling schedule (e.g., round robin, setting z equal to y), and the address of y as the intended receiver.

If z receives the RT2 correctly and has data for x, it immediately transitions to the XMIT state, where it transmits a data packet to x; otherwise, z transitions to the BACKOFF state to remain quiet for a time period that is long enough to allow the designated receiver y start sending a CTS, enabling x to send a data packet.

If y receives the RT2 correctly and does not hear any transmission from z for a period of time equal to a maximum round-trip delay with any neighbor, it transitions to the CTS state and sends a CTS addressed to x if no carrier is detected in the channel.

Any node in PASSIVE state that detects noise in the channel must transition to the BACKOFF state. Any node other than y and z receiving an RT2 transitions to the BACKOFF state. Any node other than x receiving the CTS for x transitions to the BACKOFF state.

When node x receives the CTS from y, it transitions to the XMIT state and transmits a data packet to y.

Therefore, PDMA can have two types of successful exchanges, which are illustrated in Fig. 1(a) and (b). The polled node z may have data to send to x (Fig. 1(a)) or z is idle and the intended receiver y sends a CTS successfully (Fig. 1(b)).

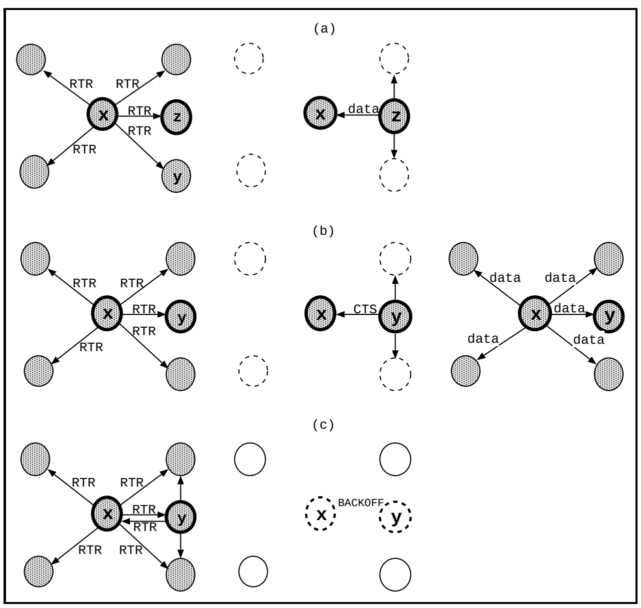

Fig. 1. PDMA illustrated

When multiple RT2s are transmitted within a one-way propagation delay a collision takes place and the nodes involved have to transition to the BACKOFF state and try again at a later time chosen at random, as shown in Fig. 1(c).

Node x determines that its RT2 was not received correctly by either z or y after a time period equal to the maximum round-trip delay to its neighbors plus turn-around times and processing delays at the nodes.

The length of RT2s and CTSs is the same and, to ensure that no data packet collides with other packets, has to be larger than the maximum propagation delay between any two neighbors.

To reduce the probability that the same nodes would compete repeatedly for the same receiver at the time of the next RT2, leading to successive collisions of RT2s, the RT2 specifies a backoff-period unit for contention. The nodes that must enter the BACKOFF state compute a random time that is a multiple of the backoff-period unit advertised in the RT2. The simplest case consists of computing a random number of backoff-period units using a uniformly distributed random variable from 1 to d, where d is the maximum number of neighbors for a receiver. The simplest backoff-period unit is the time it takes to send a small data packet successfully.

III. Floor Assignment in PDMA

Theorem 1 below shows that PDMA ensures that any packet is sent to its destination within a finite time after it becomes ready for transmission, and that there are no collisions between data packets and any other transmissions, under the following assumptions ([4]):

A0) A station transmits an RT2 that does not collide with any other transmissions with a non-zero probability.

A1) The maximum end-to-end propagation time in the channel is $\tau < \infty$.

A2) A packet sent over the channel that does not collide with other transmissions is delivered error free with a non-zero probability.

A3) All nodes execute PDMA correctly.

A4) The transmission time of an RT2 and a CTS is γ, the maximum transmission time of a data packet is δ, and the hardware transmit-to-receive transition time is $\tau \leq \epsilon$. furthermore, $2\tau < \gamma \leq \delta < \infty$.

A5) There is no capture or fading in the channel.

Theorem 1: PDMA provides correct floor acquisition in the presence of hidden terminals.

Proof: The topology depicted in Figure 2 includes all the possible situations of hidden terminals with respect to source S and receiver R. Node L represents all the neighbors of S that can be hidden from R and therefore we can have interference at S. Node K represents all the neighbors of L hidden from S that can cause interference at L so that L won't be able to follow the packet exchange between S and R. Likewise, node X is a neighbor of R that is hidden from S, causing interference at R; and node Y is a neighbor of X that is hidden from R, preventing X from understanding the dialogue between S and R.

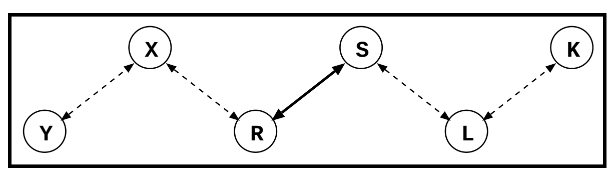

Fig. 2. PDMA floor assignment

The proof needs to show that, if node S sends a data packet to R, no other transmission can collide with it, independently of the status of all aforementioned nodes.

Node S can send a data packet to R only after receiving a successful RT2 from R. Without loss of generality, we can assume that, at time t_0, node R sends an RT2 to S. Because the channel has a minimum propagation delay larger than 0, all the neighbors of R start receiving R's RT2 at some time $t_1 > t_0$. Node X either receives the RT2 from R in the clear at time $t_2^X \leq t_0 + \gamma + \tau$ or it hears noise and must back off until $t_2^X > t_0 + 2\gamma + \delta + 5\tau + \epsilon$.

If S receives R's RT2 with errors or if a collision happens with a transmission from a node that is hidden from R (e.g., L), then S does not send a data packet; in this case, S backs off until $t_2^S > t_0 + 2\gamma + \delta + 5\tau + \epsilon$ seconds. Otherwise, S receives a clear RT2 at time t_1 and sends a data packet after waiting for τ seconds. However, this can happen only after S has transitioned to a listening mode, which occurs at time $t_2^S \leq t_0 + \gamma + \epsilon + \tau$.

At time $t_3 = t_2^S + \delta + \tau \leq t_0 + \gamma + \delta + 2\tau + \epsilon$ node R has received the data packet. Because X is in the BACKOFF state until time $t_2^X > t_0 + 2\gamma + \delta + 4\tau + \epsilon$, there are no packets sent from node X to node R; therefore, the data packet sent from S to R, must be received from R in the clear.

In the case that S receives an RT2 in the clear but does not have any data packets to send to R, S waits for 2τ, until time $t_3^S = t_0 + \gamma + \epsilon + 3\tau$. If S senses that the channel is idle at that moment, it transmits a CTS. Nodes R and L receive the CTS at time $t_4 < t_3^S + \gamma + \tau$. If K tries to transmit a packet at the same time, a collision happens and both stations back off; otherwise, K receives a clear CTS and must back off for $\gamma + \delta + 4\tau + \epsilon$ seconds, while S waits R to send any data packet that it may have. Node R always receives a clear CTS from S as node X is in the BACKOFF state, until time $t_2^X > t_0 + 2\gamma + \delta + 5\tau + \epsilon$. Finally, the data packet sent from node R to S will also be in the clear as S receives the data packet at time $t_3^S < t_0 + 2\gamma + \delta + 4\tau + \epsilon$ and L is in the CTS state until at least time $t_0 + 2\gamma + \delta + 4\tau + \epsilon$.

In the special case that we have a series of RT2s colliding in the channel, let t_0 be the time when a given node w receives the first RT2 packet of the series. In order for this to happen, all the colliding RT2s must be transmitted no later than $t_0 + \tau$; otherwise w would be able to detect the existence of an other RT2 already in the channel. If such a collision occurs, node w should be able to detect it no later than $t_0 + 2\tau$. □

IV. Approximate Throughput Analysis

We analyze the throughput of PDMA and MACA-BI ([9], [10]) using the model first introduced by Kleinrock and Tobagi [7] for CSMA protocols and used subsequently to analyze MACA [6] and FAMA [4]. According to this model the following assumptions are made for both PDMA and MACA-BI:

1. The network is fully connected and has N nodes.
2. A single unslotted channel is used for all packets, and the channel introduces no errors.
3. All nodes can detect collisions perfectly.
4. Each node has an independent Poisson source of data packets, with a mean rate of $\frac{\lambda}{N}$ data packets per second.
5. Each node has at most one data packet to send at any given time.
6. The size for a data packet is δ seconds and the size of an RTR, RT2 and an CTS is γ seconds.
7. The turn-around time ϵ is 0.
8. The propagation delay of the channel between any two nodes is τ seconds.

As we have shown in the previous section, PDMA ensures that no data packets collide with others under the above assumptions, and the same is also true for MACA-BI.

The average channel utilization is:

$$S = \frac{\overline{U}}{\overline{B} + \overline{I}} \tag{1}$$

where $\overline{B}$ is the expected duration of a busy period, defined to be a period of time during which the channel is being utilized; $\overline{I}$ is the expected duration of an idle period, defined as the time interval between two consecutive busy periods; and $\overline{U}$ is the average length of time during a busy period that the channel is used for transmitting user data successfully.

A. PDMA

The following theorem provides throughput as a function of offered load for PDMA.

Theorem 2: The throughput of PDMA is given by

$$S = \frac{\delta}{\delta + \tau + \frac{1}{\lambda} + (\gamma + 3\tau) \cdot e^{-\frac{\lambda\gamma}{N^2}} + (\gamma + 2\tau) \cdot e^{\lambda\tau}} \tag{2}$$

Proof: To analyze the throughput of PDMA as a function of the rate of data packet arrivals at nodes, we assume that all nodes have the same rate of packet arrivals and that a node chooses the recipient of a data packet with equal probability. Therefore, the arrival of data packets at a given node for a specific receiver is $\frac{\lambda}{N^2}$. Because nodes send RT2s when they obtain data packets to send, the arrival of RT2s is also a Poisson with a rate of $\frac{\lambda}{N}$ RT2s per second.

With PDMA whenever an RT2 is transmitted successfully a packet will always follow, either from the sender or the receiver. Therefore, the probability of success P_S, is equal to the probability with which an RT2 is transmitted successfully. Because all nodes are connected, an RT2 from node w is successful if there are no other RT2s transmitted within τ seconds from the start of the RT2. After this vulnerability period of τ seconds, all the nodes detect the carrier signal and act appropriately. Accordingly,

$$P_S = e^{-\lambda\tau} \tag{3}$$

According to the PDMA specification, a successful transmission occurs in two cases: (a) when the polled sender has a data packet to send, or (b) when the sender has nothing to send and the intended receiver sends a CTS successfully. The probability, P_{S1}, with which the first case happens is equal to the probability that an RT2 is sent in the clear, times the probability that there is at least one packet arrival at the polled node destined to the polling node during γ seconds, that is:

$$P_{S1} = e^{-\lambda\tau}(1 - e^{-\frac{\lambda\gamma}{N^2}}) \tag{4}$$

The time that it takes to transmit a successful packet in this case is equal to the transmission time for a successful RT2 and the data packet, plus the propagation time associated with each; that is:

$$T_{S1} = \gamma + \delta + 2\tau \tag{5}$$

The second case of a successful busy period happens when the polled sender does not have a packet to send and therefore it sends a CTS packet back to the sender of the RT2 enabling the node to send a data packet. The probability that this scenario happens is equal to the probability that an RT2 is sent in the clear, times the probability that the polled sender experiences no arrivals of data packets destined for the polling node in γ seconds; that is:

$$P_{S2} = e^{-\lambda\tau} e^{-\frac{\lambda\gamma}{N^2}} \tag{6}$$

The time for a successful data packet in this case, T_{S2} is equal to the transmission time for a successful RT2, CTS and data packet, plus the waiting time of 2τ needed for the polled sender, plus the corresponding propagation delays; that is: $T_{S2} = 2\gamma + \delta + 5\tau$.

A busy period always consists of at least an RT2 and the associated propagation delay, i.e. $\gamma + \tau$. If the RT2 fails, the busy period lasts $\gamma + \tau + \overline{Y}$, where $\overline{Y}$ is the time between the start of the first and the last RT2s of the busy period and is the same as in CSMA [11]. Accordingly, a failed busy period lasts

$$T_F = \gamma + 2\tau - \frac{1 - e^{-\lambda\tau}}{\lambda} \tag{7}$$

If a busy period is successful because the polled source has a packet to send, the busy period lasts $\gamma + \delta + 2\tau$, which corresponds to the duration of the RT2, the data packet and their propagation delays. Alternatively, if the busy period is successful when the polled source is silent, it lasts $2\gamma + \delta + 5\tau$, which corresponds to an RTR, a CTS, a data packet, their propagation delays, and 2τ wait time before the CTS is sent. From the above, it follows that the duration of the average busy period is given by

$$\begin{aligned}\overline{B} &= \gamma + 2\tau - \frac{1 - e^{-\lambda\tau}}{\lambda} \\ &+ e^{-\lambda\tau} \cdot \left[(1 - e^{-\frac{\lambda\gamma}{N^2}}) \cdot (\delta + \tau) + e^{-\frac{\lambda\gamma}{N^2}} \cdot (\gamma + \delta + 4\tau)\right] \\ &= \gamma + 2\tau - \frac{1}{\lambda} + e^{-\lambda\tau} \cdot \left[\delta + \tau + \frac{1}{\lambda} + e^{-\frac{\lambda\gamma}{N^2}} \cdot (\gamma + 3\tau)\right] \quad (8)\end{aligned}$$

When PDMA is used, the channel is only idle for a time period equal to the interarrival rate of RT2s, and $\overline{I} = \frac{1}{\lambda}$. The average utilization time at node w is the proportion of time in which useful data are sent, consequently, $\overline{U} = \delta \cdot P_S = \delta \cdot e^{-\lambda\tau}$.

Equation (2) follows from substituting $\overline{I}$, $\overline{U}$ and Eq. (8) into Eq. (1). □

B. MACA-BI

MACA-BI is very similar to PDMA, with the exception that, whenever the sender polled by an RTR has no packet to send to the polling node, no special action is taken. The performance analysis for MACA-BI as presented in [9] does not take into account the probability that an RTR sent in the clear may not result in a data packet being sent. Accordingly, the performance results shown for MACA-BI by [9] correspond to the case in which the receiver knows exactly when a neighbor has packet for it, and is therefore an unattainable upper bound on performance.

Our analysis of MACA-BI makes the same assumptions used for PDMA, assumes that an RTR has the same length as an RT2, and that a node sends an RTR to a neighbor when the node has data to send.

Theorem 3: The throughput for the MACA-BI protocol, is given by

$$S = \frac{\delta \cdot (1 - e^{-\frac{\lambda\gamma}{N^2}})}{\delta + \tau + \frac{1}{\lambda} + (\tau - \delta) \cdot e^{-\frac{\lambda\gamma}{N^2}} + (\gamma + 2\tau) \cdot e^{\lambda\tau}} \quad (9)$$

Proof: The probability that a successful transmission occurs is equal to the probability that an RT2 is transmitted successfully times the probability that there is at least one data packet arrival at the sender for the receiver within γ seconds, that is,

$$P_S = e^{-\lambda\tau} \cdot (1 - e^{-\frac{\lambda\gamma}{N^2}}) \quad (10)$$

In this case the duration of the busy period is $\gamma + \delta + 2\tau$.

In MACA-BI, a failed busy period can occur in two cases: (a) when there is a collision between RTRs, and (b) when an RTR is sent in the clear but the polled sender does not have a data packet to send. The first case occurs with probability:

$$P_{F1} = 1 - e^{-\lambda\tau} \quad (11)$$

The duration of such a failed busy period is the same as Eq. 7. The probability of the second busy period scenario occurring is given by

$$P_{F2} = e^{-\lambda\tau} \cdot e^{-\frac{\lambda\gamma}{N^2}} \quad (12)$$

and the duration of such a busy period is $\gamma + 3\tau$, corresponding to the RTR, the propagation delay, and the time nodes must wait after receiving the RTR in the clear.

Therefore, the length of the average busy period is given by

$$\begin{aligned}\overline{B} &= \gamma + 2\tau - \frac{1 - e^{-\lambda\tau}}{\lambda} \\ &+ e^{-\lambda\tau} \cdot (1 - e^{-\frac{\lambda\gamma}{N^2}}) \cdot (\delta + \tau) + e^{-\lambda\tau} \cdot e^{-\frac{\lambda\gamma}{N^2}} \cdot (2\tau) \\ &= \gamma + 2\tau - \frac{1}{\lambda} + e^{-\lambda\tau} \cdot \left[\delta + \tau + \frac{1}{\lambda} + (\tau - \delta) \cdot e^{-\frac{\lambda\gamma}{N^2}}\right] \quad (13)\end{aligned}$$

The length of the average idle period is $\frac{1}{\lambda}$ as in PDMA, and the length of the average utilization period is

$$\overline{U} = \delta \cdot P_S = \delta \cdot e^{-\lambda\tau} \cdot (1 - e^{-\frac{\lambda\gamma}{N^2}}) \quad (14)$$

The theorem follows by substituting the values of the average idle, busy and utilization periods in Eq. (1). □

C. Performance Comparison

To compare PDMA with other widely known MAC protocols, we introduce the following variables:

$$\begin{aligned} a &= \frac{\tau}{\delta} \text{(normalized propagation delay)} \\ b &= \frac{\gamma}{\delta} \text{(normalized control packets)} \\ c &= \frac{\varepsilon}{\delta} \text{(normalized transmit to receive turn around time)} \\ G &= \lambda \times \delta \text{(Offered Load, normalized to data packets)} \end{aligned}$$

With the above notation, we calculate the normalized throughput for PDMA, MACA-BI [9], non-persistent CSMA [7], MACA [6], and FAMA-NCS [4] as shown in table I.

In our comparison, we assume a fully-connected network topology with a propagation delay of $1\mu s$; 500 and 1000 byte data packets; a length of 20 bytes for RTRs, RT2s and CTSs for PDMA; CTSs of length $\gamma + \tau$ for FAMA-NCS; a channel data rate of 1 Mb/s; and zero preamble and processing overhead for convenience.

The performance demonstrated by PDMA is much better than MACA and MACA-BI. The upper bound for non-persistent CSMA, which assumes an ideal channel over which acknowledgments to packets are sent in zero time [7], has a similar throughput curve as PDMA coming short only when the offered load is between 100 and 2000 for a fully-connected topology. The number of nodes in the network does not affect the throughput achieved by CSMA, FAMA-NCS and PDMA protocols.

As we can see in Figures 3 and 4, for a fully-connected network with 500 and 1000 bytes data packets PDMA and FAMA-NCS show very similar throughput curves in all experiments. The throughput in FAMA-NCS is always less than or equal to the one achieved with PDMA.

V. CONCLUSIONS

We have specified and analyzed the receiver-initiated multiple access (PDMA) protocol. We compared PDMA with non-persistent CSMA, MACA, MACA-BI and FAMA-NCS, and showed that the throughput in PDMA is much higher than MACA, always better than FAMA-NCS and MACA-BI, and better than non-persistent CSMA at high loads. We also argued that the average packet delay experienced with PDMA is less than that with CSMA. As such PDMA is the first MAC protocol based on a receiver-initiated collision avoidance mechanism that outperforms MAC protocols for wireless networks based on sender-initiated collision avoidance. Our work continues to extend the analysis of PDMA for the case of a multihop network.

Protocol	*Unslotted version*
np-CSMA [7]	$\frac{Ge^{-aG}}{G(1+2a)+e^{-aG}}$
MACA [6]	$\frac{1}{e^{(2b+a)G}(b+a+\frac{1}{G}+F')+e^{bG}\left[b+\frac{a}{2}+P'(a-F')\right]+1+\frac{3a}{2}+F'+P'(a-F')}$ where $F' = \frac{e^{bG}-1-bG}{bG(1-e^{-bG})}$ and $P' = \frac{e^{-bG}-e^{-G(a+b)}}{1-e^{-G(a+b)}}$
FAMA-NCS	$\frac{1}{b+4a+1+\frac{1}{G}+e^{aG}(b+4*a)}$
MACA-BI	$\frac{1-e^{-\frac{bG}{N^2}}}{1+\frac{1}{G}-e^{-\frac{bG}{N^2}}+(b+2a)e^{aG}}$
PDMA	$\frac{1}{1+a+\frac{1}{G}+(b+3a)\cdot e^{-\frac{bG}{N^2}}+(b+2a)\cdot e^{aG}}$

TABLE I

THROUGHPUT OF SENDER-INITIATED AND RECEIVER-INITIATED MAC PROTOCOLS

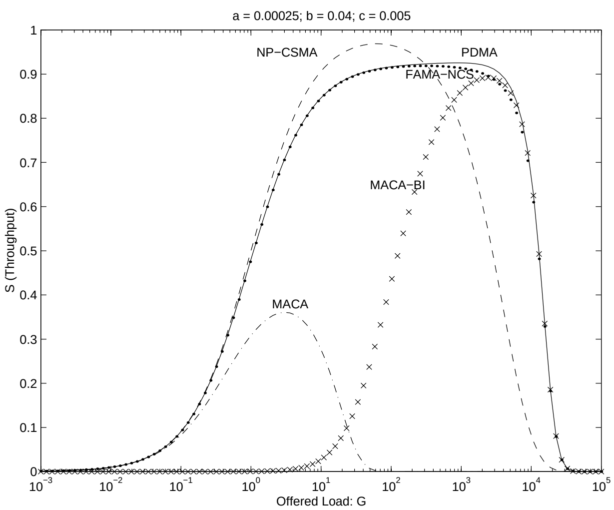

Fig. 3. 500 Byte data packets; 1Mbit/sec network speed

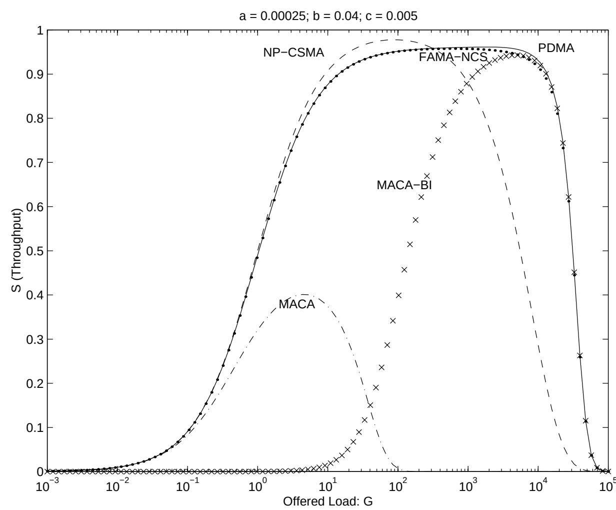

Fig. 4. 1000 Byte data packets; 1Mbit/sec network speed

REFERENCES

[1] V. Bharghavan, A. Demers, S. Shenker and L. Zhang, “MACAW: A Media Access Protocol for Wireless LAN's” *Proc. ACM SIGCOMM 94*, London, UK, Aug. 31 - Sep. 2, 1994.

[2] C. L. Fullmer and J.J. Garcia-Luna-Aceves, “Floor Acquisition Multiple Access for Packet-Radio Networks,” *Proc. ACM SIGCOMM 95*, Cambridge, MA, August 30-September 1, 1995.

[3] C. L. Fullmer and J.J. Garcia-Luna-Aceves, “Solutions to Hidden Terminal Problems in Wireless Networks,” *Proc. ACM SIGCOMM 97*, Cannes, France, September 14-18, 1997.

[4] *P802.11–Unapproved Draft: Wireless LAN Medium Access Control (MAC) and Physical Specifications*, IEEE, January 1996.

[5] P. Karn, “MACA - a new channel access method for packet radio,” in *ARRL/CRRL Amateur Radio 9th Computer Networking Conference*, pp. 134–40, ARRL, 1990.

[6] L. Kleinrock and F. A. Tobagi, “Packet switching in radio channels: Part I - carrier sense multiple-access modes and their throughput-delay characteristics,” *IEEE Trans. Commun.*, vol. COM-23, no. 12, pp. 1400–1416, 1975.

[7] R. Rom and M. Sidi, “Multiple Access Protocols Performance and Analysis,” *Springer-Verlag*, 1990.

[8] F. Talucci, M. Gerla and L. Fratta “MACA-BI: A Reveiver Oriented Access Protocol for Wireless Multihop Networks,” *Proc. of IEEE PIMRC*, 1997.

[9] F. Talucci and M. Gerla “MACA-BI(MACA by invitation). A Wireless MAC Protocol for High Speed Ad hoc Networking,” *Proc. of IEEE ICUPC*, 1997.

[10] 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,” *IEEE Trans. Commun.*, vol. COM-23, no. 12, pp. 1417–1433, 1975.

[11] F. A. Tobagi and L. Kleinrock, “Packet switching in radio channels: Part III - polling and (dynamic) split-channel reservation multiple access,” *IEEE Trans. Commun.*, vol. COM-24, no. 8, pp. 832–845, 1976.