

An Adaptive Power Controlled MAC Protocol for Wireless Ad Hoc Networks

Pan Li, Xiaojun Geng, and Yuguang Fang

Abstract—Transmission power control (TPC) has been extensively used not only to save energy, but also to improve the network throughput in wireless ad hoc networks. Among the existing throughput-oriented TPC protocols, many can achieve significant throughput improvement but have to use multiple channels and/or multiple transceivers, and others just require a single channel and a single transceiver but can only have limited throughput enhancement. In this paper, we propose a new adaptive transmission power control protocol, ATPMAC, which can improve the network throughput significantly using a single channel and a single transceiver. Specifically, by controlling the transmission power, ATPMAC can enable several concurrent transmissions without interfering with each other. Moreover, ATPMAC does not introduce any additional signalling overhead. We show by simulations that ATPMAC can improve the network throughput by up to 136% compared to IEEE 802.11 in a random topology.

Index Terms—Wireless ad hoc networks; MAC protocol; transmission power control.

I. INTRODUCTION

A WIRELESS ad hoc network is a network where nodes communicate with each other via wireless medium directly or indirectly with the help of other nodes. It has gained popularity recently due to its easy and quick deployment with low cost. In ad hoc networks, the wireless channel is shared by all the nodes, and hence a medium access control (MAC) protocol is needed to coordinate their transmissions to reduce the collision. Although it was initially standardized for wireless local area networks (WLANs), IEEE 802.11 DCF (Distributed Coordination Function), known as Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA), with an optional use of RTS/CTS [2], is now also widely used as the MAC protocol in wireless ad hoc networks.

However, the IEEE 802.11 MAC has two main disadvantages when used in ad hoc networks. *First*, energy is used inefficiently. IEEE 802.11 MAC uses the same transmission power for all nodes to transmit all their packets, no matter how

close a transmitter may be to its intended receiver. *Second*, the spatial reuse of the network is low. According to IEEE 802.11 MAC, when one node is transmitting, the other nodes in its physical carrier sensing range should keep silent to avoid interference. Thus, even those transmissions that will not interfere with the ongoing one are still blocked.

Since in some cases the nodes in ad hoc networks have limited power, energy is indeed a very important issue. Many protocols are proposed to save the power consumption, like those in [6] [7] [8]. In these protocols, nodes transmit RTS/CTS at the same maximum power, and transmit DATA/ACK at the minimum power needed for successful transmission, which depends on the link distance, as well as the interference level at the receiver. By doing this, the transmission power can be saved. Unfortunately, the spatial reuse is pretty low, and many collisions happen between the control packets (RTS/CTS) and the data packets (DATA/ACK) due to different physical carrier sensing ranges. Thus, the network throughput cannot be improved.

On the other hand, there are also some papers focusing on the throughput enhancement. [9] [10] propose to use two channels, and two transceivers, to improve the network throughput. Significant improvements over IEEE 802.11 MAC are observed in simulations. However, the use of multi-channel and multi-transceiver introduces additional hardware cost and implementation complexity.

Recently, Jia *et al.* [5] and Ding *et al.* [4] propose δ -PCS and DEMAC, respectively, to improve the network throughput using a single channel and a single transceiver. These two protocols try to adjust the transmission power for each packet so that the transmission can be successful and at the same time it will not cause too much interference to other transmissions. Since δ -PCS and DEMAC still work under the same decision rule as IEEE 802.11, they only achieve limited improvement.

Muqattash and Krunch [11] also propose in a throughput-oriented MAC protocol utilizing a single channel and a single transceiver, called POWMAC. Different from the above protocols, POWMAC uses a new decision rule: when a node overhears other nodes' transmissions, it is still allowed to carry out its own DATA transmission as long as it does not interfere with the ongoing ones. Thus, according to POWMAC, several transmissions can happen concurrently.

However, POWMAC cannot gain dramatic improvement on network throughput due to the following two reasons. First, it introduces additional signalling overhead. Specifically, $N(N > 1)$ concurrent transmissions require N RTS/CTS exchanges. This overhead gets more serious when the channel rate for data transmission increases. Second, several concur-

Manuscript received September 7, 2008; revised December 24, 2007; accepted January 10, 2008. The associate editor coordinating the review of this paper and approving it for publication was T. Hou.

This work was supported in part by the U.S. National Science Foundation under grants CNS-0721744 and DBI-0529012. The work of Fang was also supported in part by the National Science Council (NSC), R.O.C., under the NSC Visiting Professorship with contract number NSC-96-2811-E-002-010 and Chungwa Telecom M-Taiwan Program M-Taoyuan Project under Contract Number 97SC06.

P. Li and Y. Fang are with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, Florida 32611-6130 (e-mail: lipanleo@ufl.edu, fang@ece.ufl.edu).

X. Geng is with the Department of Electrical and Computer Engineering, California State University, Northridge, CA 91330 (e-mail: xj-geng@csun.edu).

Digital Object Identifier 10.1109/T-WC.2009.070993

rent DATA transmissions may not take place if they are not synchronized due to the existence of propagation delay. We will discuss this further later.

In this paper, we propose a new adaptive transmission power controlled MAC protocol, called ATPMAC, to enhance the network throughput using a single channel and a single transceiver. ATPMAC adopts the same decision rule as POWMAC to enable concurrent transmissions. But, ATPMAC does not incur any additional signalling overhead, i.e., only one RTS/CTS exchange for $N(N > 1)$ concurrent transmissions. Moreover, ATPMAC provides two solutions to the synchronization problem mentioned above so that the concurrent DATA transmissions can happen even though the propagation delay exists. By doing this, ATPMAC can achieve significant throughput improvement over IEEE 802.11, which is up to 136% in a random topology as we will show by simulations.

The rest of this paper is organized as follows. In Section II, we briefly introduce the operations of IEEE 802.11 DCF protocol and the power propagation model. Section III details our proposed ATPMAC protocol. Some simulation results are shown in Section IV. We finally conclude this paper in section V.

II. PRELIMINARIES

A. IEEE 802.11 MAC

The fundamental access method of the IEEE 802.11 MAC is a DCF (Distributed Coordination Function) known as Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) with an option of RTS/CTS. The four-way handshake procedure (RTS/CTS/DATA/ACK), which is used to deal with the hidden terminal problem, is as follows. Before a node begins to transmit, it should first sense the channel to determine whether there is any ongoing transmission. If the channel is busy, the node shall defer until the channel is sensed idle for a period of DIFS. Then the node randomly chooses a backoff period according to the contention window and starts a backoff timer to backoff. The backoff timer decreases by 1 after the channel is idle for the duration of a slot. If the channel is sensed busy during any slot in the backoff interval, the backoff timer will be suspended. It can be resumed only after the channel is idle for a period of DIFS again. After the backoff timer reduces to 0, the sender transmits a RTS omnidirectionally. After correctly receiving the RTS, the receiver responds with a CTS after a period of SIFS. Similarly, after correctly receiving the CTS, the sender begins to transmit the data a period of SIFS later. This transmission ends after the receiver correctly receives the data and responds with an ACK. All four kinds of frames contain an estimated duration of the rest time of the transmission. Other nodes that receive these frames update their NAVs (Network Allocation Vector) with the duration. Every NAV decreases by 1 after a time slot. Those nodes are only allowed to transmit after they sense the channel idle for a period of DIFS after their NAVs expire.

B. Power Propagation Models

The power propagation models are used to predict received signal strength. A general model [13] is given as follows:

$$P_r(d) = P_t h(G_t, G_r, h_t, h_r, L, \lambda) \frac{1}{d^\gamma} \quad (1)$$

where P_t and P_r are the transmitted power and the received power, respectively, G_t and G_r are the gain factors for the transmitter antenna and the receiver antenna, respectively, h_t and h_r are the antenna heights of the transmitter and the receiver, respectively, d is the distance between transmitter and receiver, L is the system loss factor not related to propagation ($L \geq 1$), λ is the wavelength, $h(\cdot)$ is a function, and γ is the path loss exponent.

III. THE PROPOSED MAC PROTOCOL: ATPMAC

According to IEEE 802.11 MAC protocol, every node has to carry out the physical carrier sensing before transmitting RTS, CTS, or DATA packets (but not ACK packets). If the channel is sensed to be busy, then the nodes cannot transmit those packets. As a result, the spatial reuse is pretty low because each time the channel can be used by only one pair of transmitter and receiver, even though some other transmissions may not interrupt the ongoing transmission. The exposed terminal problem can be such an example.

In this paper, we propose a single-radio, single-channel, and single-rate MAC protocol to improve the spatial reuse by controlling the transmission power so that several transmissions can be allowed at the same time without interfering with each other. This is an adaptive transmission power control MAC protocol, which we call ATPMAC. The idea here is that a new transmission can still be allowed as long as it does not interfere with the ongoing transmission.

ATPMAC does not use any new control packets other than RTS and CTS. Neither does it incur any other signalling overhead than one RTS/CTS handshake before a DATA transmission. Instead, one RTS/CTS handshake can be followed by several concurrent DATA transmissions, which do not interfere with each other. Specifically, in ATPMAC, the nodes that overhear RTS or CTS can make a decision on whether they can transmit DATA packets to their intended receivers based on some useful information carried by RTS/CTS.

We will first introduce a table maintained by each node in Section III-A, and then explain in details how ATPMAC works when a node overhears CTS and RTS, respectively, in Section III-B and Section III-C. After that, we will introduce how we tune the physical carrier sensing threshold in order for ATPMAC to work more effectively in Section III-D, and give some more discussions in Section III-E, respectively. We also compare ATPMAC with POWMAC at the end of this section.

A. A Table Maintained by Each Node

As shown in Table I, each node maintains a table to keep some information of their neighboring nodes. "Node ID" is the MAC address of a neighboring node. "Min Power", denoted by P_{min} , is the minimum transmission power required to successfully send a packet to that neighboring node when it does not suffer from any other interferences. "Max Power", denoted by P_{max} , is the maximum transmission power allowed for the current node keeping this table to transmit packets when that neighboring node is engaged in one transmission. "NAV" is the time that the neighboring node will finish its ongoing transmission. Each time a node overhears a packet from one of its neighboring nodes, it updates this table. The details will be introduced later.

TABLE I
SOME INFORMATION OF NEIGHBORING NODES.

Node ID	Min Power (P_{min})	Max Power (P_{max})	NAV
---------	-------------------------	-------------------------	-----

B. Overhearing CTS

In this subsection, we introduce how ATPMAC works when a node overhears CTS.

When a receiver j receives RTS from a transmitter i , it can detect the reception power P_r^i , and obtain the transmission power P_t^i , which is a new field we add in RTS frames. According to the power propagation model in (1), we have

$$P_r^i = C \cdot \frac{P_t^i}{d_{ij}^\gamma}, \quad (2)$$

where d_{ij} is the distance between node i and node j , and C is a constant.

Denote the receiver sensitivity by RX_{th} . Then, by assuming the physical channel is symmetric, the minimum power required for the receiver j to successfully transmit a packet to the transmitter i , i.e., P_{min}^i as mentioned before, satisfies

$$RX_{th} = C \cdot \frac{P_{min}^i}{d_{ij}^\gamma}. \quad (3)$$

From (2) and (3), we can get

$$P_{min}^i = \frac{P_t^i \cdot RX_{th}}{P_r^i}. \quad (4)$$

After obtaining P_{min}^i , the receiver j should check in Table I to find those active neighboring nodes denoted by set S , i.e.,

$$S = \{k \mid NAV^k > t_{now}\}$$

where NAV^k is the time that neighboring node k will finish its ongoing transmission, and t_{now} is the current time. Then, the maximum allowed transmission power of the receiver j , denoted by P_{allow}^j , is

$$P_{allow}^j = \begin{cases} \min_{k \in S} \{P_{max}^k\} & \text{if } S \neq \emptyset \\ P_{MAX} & \text{if } S = \emptyset \end{cases} \quad (5)$$

where P_{max}^k is the maximum transmission power of node k at which j 's transmission will not interfere with k 's, \emptyset stands for the empty set, and P_{MAX} is the maximum allowed transmission power of the nodes. We will present how to obtain P_{max}^k shortly.

If P_{allow}^j is less than P_{min}^i , then the receiver j is not allowed to reply with CTS because this CTS will definitely not be received by the transmitter. Otherwise, CTS is transmitted after a period of SIFS with the transmission power P_{allow}^j . Thus, this CTS transmission will not interfere with j 's active neighboring nodes' transmissions, and it is possible that this CTS could be correctly received.

The same as RTS frame, our CTS frame also contains its transmission power. Since the CTS frame defined in IEEE 802.11 standard only has the MAC address of the frame's receiver, we add a new field called "Transmitter Address" in our CTS frame to put in the MAC address of the frame's transmitter. By doing this, other nodes overhearing CTS from the receiver j can update their information about j kept in Table I. We will introduce this process later.

Besides, we also add another new field called "Interference Level" in our CTS frame, which is the maximum average interference level each neighboring node is allowed to generate to receiver j . Denote "Interference Level" by P_{interf} . We can obtain

$$\begin{aligned} P_{interf}^j &= \frac{P_r^i}{SINR} - P_{noise} \\ &= \frac{P_r^i - SINR \cdot P_{noise}}{N \cdot (1 + \beta)} \\ &= \frac{P_r^i - SINR \cdot P_{noise}}{N \cdot (1 + \beta) \cdot SINR} \end{aligned} \quad (6)$$

where SINR is the signal-to-interference and noise ratio required to support a certain data rate (SINR is a constant since we do not consider rate adaptation here), N is the number of the neighboring nodes of the receiver j , which can be obtained by checking the number of nodes in Table I, and β ($\beta > 0$) indicates the interference caused by the nodes out of the transmission range, which is about 0.5 for the two-ray propagation model and uniformly distributed terminals [14].

After a CTS is sent out, some neighboring nodes of the receiver j may overhear it and hence can update their information about node j . P_{min}^j is calculated similar to (4), i.e.,

$$P_{min}^j = \frac{P_t^j \cdot RX_{th}}{P_r^j}. \quad (7)$$

Then, next time when a neighboring node k wants to send packets to node j , it can carry out the transmission only if its maximum allowed transmission power, P_{allow}^k , is no smaller than P_{min}^j .

Since this CTS contains the transmission power of receiver j , denoted by P_t^j , for a neighboring node k , we have

$$P_r^j = C \cdot \frac{P_t^j}{d_{jk}^\gamma}, \quad (8)$$

and

$$P_{interf} = C \cdot \frac{P_{max}^j}{d_{jk}^\gamma}. \quad (9)$$

where d_{jk} denotes the distance between receiver j and the neighboring node k , and P_{max}^j is the maximum transmission power allowed for node k to transmit packets without affecting the reception of the following DATA packets at receiver j . From (8) and (9), we obtain

$$P_{max}^j = \frac{P_{interf} \cdot P_t^j}{P_r^j}. \quad (10)$$

After successfully overhearing the CTS from node j , a neighboring node k will update the NAV field in Table I for node j . If node k does not want to send out any packets, it does not set its NAV. With P_{allow}^j defined in (5), even later it has some packets to transmit, those transmissions will not interfere with j 's reception. Or, if it has a DATA packet for node j , node k will set its NAV in the same way as that defined in IEEE 802.11 standard. If the neighboring node k has a DATA packet for some node l other than node j , it will also set its NAV if $P_{allow}^k < P_{min}^l$. It can carry out the transmission a period of SIFS later only if the maximum transmission power of node k is no smaller than the minimum transmission power required to transmit packets to node l . Thus, there is a good chance that some neighboring nodes of receiver j can transmit

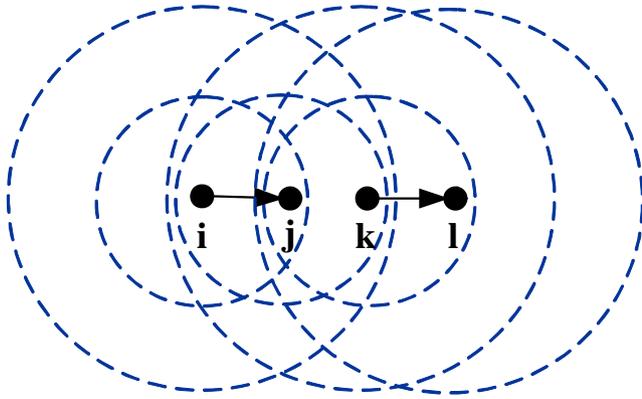


Fig. 1. An example that two concurrent transmissions happen after a CTS is overheard. The big circle is the carrier sensing range, and the small circle is the transmission range.

DATA packets at the same time as node i , and the spatial reuse can be highly improved.

For example, as shown in Fig. 1, node j is both in the transmission range of node i and in that of node k ; node k is outside the transmission range, but within the carrier sensing range of node i ; node l is within the transmission range of node k , and outside the carrier sensing range of node i . Assume there are two flows, one from node i to node j , and the other from node k to node l . According to IEEE 802.11, there can be only one transmission at a time. However, according to our proposed ATPMAC, these two flows may happen at the same time after node k overhears a CTS from node j .

Moreover, it is possible that different transmissions have DATA packets of different lengths. Should they be carried out concurrently? We contend that they should still be allowed to. As shown in Fig. 2, there are three transmissions beginning at the same time. The DATA packet of transmission 1 is longer than that of transmission 2, and shorter than that of transmission 3. However, all the three DATA packets are still able to be followed by ACK packets since the receiver does not need to do the physical carrier sensing before transmitting the ACK packets.

Besides, if a neighboring node k of receiver j is allowed to transmit a DATA packet to node l , it is still possible that node l cannot successfully receive the DATA packet, or node k cannot successfully receive the ACK packet. If node k performs according to IEEE 802.11, it will double its contention window and retransmit the data again. This may lead to the starvation of node k if node i and node j keep using the channel, which is unfair. As a result, in our MAC protocol, node k does not double its contention window if this DATA transmission is not successful. Thus, after the transmission of node i and node j terminates, node k can contend for the channel with node i and j fairly.

C. Overhearing RTS

Next, we introduce how ATPMAC works after a node overhears RTS.

We add a new field in our ACK frames called ‘‘Transmission Power’’ to put in the transmission power P_t^j of the ACK packets. So, when node i receives an ACK from node j , it can obtain the reception power P_r^j , as well as the transmission

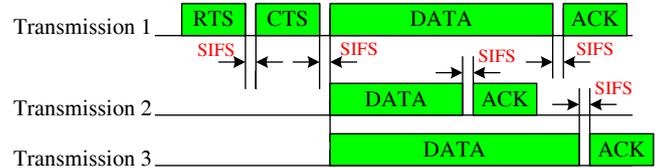


Fig. 2. An example that three concurrent transmissions have DATA packets of different lengths.

power of the ACK packet. Thus, node i can calculate the maximum average interference level each neighboring node is allowed to node i , denoted by P_{interf}^i , in a way similar to (6), i.e.,

$$P_{interf}^i = \frac{P_r^j - SINR \cdot P_{noise}}{N \cdot (1 + \beta) \cdot SINR}.$$

Besides, node i also updates P_{min}^j in Table I according to (7).

Next time, when node i has a RTS packet to transmit, it will put P_t^i and P_{interf}^i in two new fields of the RTS frame, respectively, i.e., ‘‘Transmission Power’’ and ‘‘Interference Level’’. Any neighboring node that overhears this RTS will update their P_{min}^i , P_{max}^i , and NAV^i in Table I accordingly. After successfully overhearing the RTS packet from node i , if a neighboring node k does not want to send out any packets, it will not set its NAV. Or, if it has a DATA packet for node i , node k will set its NAV in the same way as that defined in IEEE 802.11 standard. If the neighboring node k has a DATA packet for some node l other than node j , it will also set its NAV if $P_{allow}^k < P_{min}^l$. It can carry out the transmission a period of $2 * SIFS + T_{CTS}$ later only if the maximum transmission power of node k is no lower than the minimum transmission power required to transmit packets to node l . Thus, there is a good chance that some neighboring nodes of transmitter i can transmit DATA packets at the same time as node i , and the spatial reuse can be highly improved.

For example, as shown in Fig. 3, node j and node l are both in the transmission range of node k ; node i is in the transmission range of node j , but outside the carrier sensing range of node k ; node l is in the transmission range of node k , but outside the carrier sensing range of node j . Assume there are two flows, one from node j to node i , and the other from node k to node l . According to IEEE 802.11, node j and node k will fairly contend with each other for the channel and there is only one transmission at a time. However, according to our proposed ATPMAC, two DATA transmissions from node j and node k , respectively, may happen at the same time after one node overhears a RTS from the other, and two ACKs from node i and node l , respectively, may also both be received successfully.

Besides, as we have explained in Section III-B, even if several transmissions have DATA packets of different lengths, they are still allowed to happen at the same time; even if the DATA transmission of a neighboring node k (or j) overhearing a RTS fails, node k (or j) does not double its contention window.

Furthermore, let us consider a special case when both overhearing a RTS and overhearing a CTS occur to the same node which plans to have a DATA transmission. Assume a node overhears a RTS packet and is allowed to carry out DATA

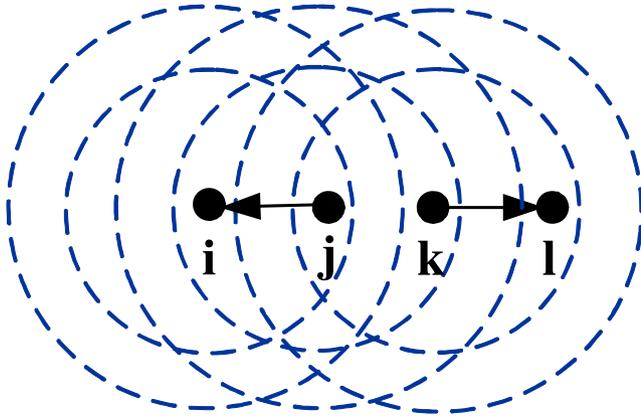


Fig. 3. An example that two concurrent transmissions happen after a RTS is overheard. The big circle is the carrier sensing range, and the small circle is the transmission range.

transmission. If later this node overhears a CTS packet and is not allowed to carry out the transmission any more, then it does not carry out the transmission as planned and waits until the channel is idle without doubling its contention window. Again, this is for the fairness issue.

D. Tuning the Physical Carrier Sensing Threshold

IEEE 802.11 standard defines two important concepts: *transmission range* and *physical carrier sensing range*, which are determined by receiver sensitivity and physical carrier sensing threshold, respectively. Two nodes within the transmission range of each other can communicate directly, and two nodes within the physical carrier sensing range of each other cannot transmit packets at the same time.

As shown in [15], physical carrier sensing range has a great impact on the network throughput. On one hand, the increase of physical carrier sensing range can alleviate the hidden terminal problem, which helps increase the throughput. On the other hand, as physical carrier sensing range increases, the spatial reuse decreases, which impairs the throughput. As a result, there exists an optimal physical carrier sensing range with respect to a certain transmission range, which is usually larger than the transmission range.

Certainly, this is true for wireless networks using IEEE 802.11. However, with respect to our proposed ATPMAC, we contend that this is not necessarily the case. As shown in Fig. 1, if node k wins the contention with node i for the channel, it will start the four-way handshake (RTS/CTS/DATA/ACK) with node l . Since node j is in the carrier sensing range of node l , it will keep silent for a period of *EIFS* after the ACK from node l is received by node k . Thus, when node i attempts to send RTS to node j , it cannot respond with CTS, which is the *receiver blocking problem*. Due to this problem, ATPMAC cannot increase the throughput much because it relies on node j 's CTS to schedule the concurrent transmissions.

To address this problem, we propose to set physical carrier sensing threshold equal to receiver sensitivity such that the carrier sensing range is the same as the transmission range. Thus, in Fig. 1, if node j successfully receives a RTS from node i , it can still reply with a CTS even if node l is transmitting an ACK to node k . Besides, we notice that node



Fig. 4. An example that two transmissions in Fig. 1 are partially overlapping.

k becomes a hidden terminal to the transmission from node i to node j . However, if node k does not transmit RTS when node j is receiving RTS from node i , then node k will receive CTS from node j and begins to transmit DATA packet at the same time as node i . If node k transmits RTS when node j is receiving RTS from node i , its transmission will not interfere with node j 's reception if node k is a little bit further away from node j . In this case, the transmission from node i to node j and that from node k to node l can be *partially overlapping*, as shown in Fig. 4.

Moreover, there would be another problem if the physical carrier sensing range is larger than the transmission range. For example, as shown in Fig. 3, assume node l is within the carrier sensing range of node j . If node k wins the channel and transmits RTS to node l , node j can correctly receive this packet and plans to transmit DATA packet at the same time as node k . However, a period of *SIFS* later node j will overhear the CTS from node l , and hence will set its NAV with a length of *EIFS* after the CTS transmission is finished, which prevents it from transmitting DATA packet to node i . We call this problem the *transmitter blocking problem*. Similarly, if node i is within the carrier sensing range of node k , node k cannot transmit its DATA packet as it has planned after node k first receives RTS from node j , and then overhears CTS from node i . However, by setting the physical carrier sensing range to the same as the transmission range, this problem can be overcome.

E. More Discussions

As we discussed in Section III-B and Section III-C, one node overhearing RTS or CTS will wait for a period of $T_{CTS} + 2 * SIFS$ or *SIFS*, respectively, to initiate a DATA transmission. However, taking the propagation delay into consideration, we need to recalculate this *waiting time*. Otherwise, our proposed ATPMAC may not work effectively.

For example, there are two concurrent transmissions in Fig. 5. Due to different propagation delays, if the DATA packet from node k arrives earlier at node j than that from node i , then node j will start to receive node k 's DATA packet instead of node i 's. Even if the reception power of the DATA packet from node i is much higher than that of the DATA packet from node k , node j still cannot correctly receive from node i according to IEEE 802.11 standard. In this case, node j cannot receive any packet. The same thing happens to node l if node i 's DATA packet arrives at node l a little bit earlier than node k 's DATA packet.

We find that there are two solutions to this problem due to the imperfect synchronization between concurrent DATA transmissions.

Solution One: We can address this problem if every node knows the exact locations of their two-hop neighboring nodes. Again, there are two cases:

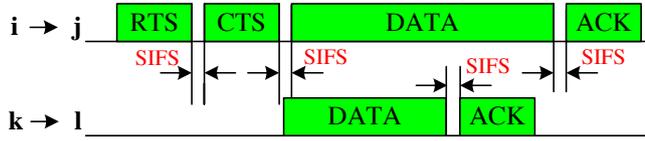


Fig. 5. A problem when we consider the propagation delay.

- **Overhearing CTS:** As shown in Fig. 1, if node k overhears CTS from node j , then, in order for node j and node l to be able to receive the DATA packet from node i and node k , respectively, we need

$$t_{ji} + SIFS + t_{ij} \leq t_{jk} + T_{wait}^k + t_{kj}$$

and

$$t_{jk} + T_{wait}^k + t_{kl} \leq t_{ji} + SIFS + t_{il}$$

where T_{wait}^k is node k 's waiting time before it transmits its DATA packet after overhearing a CTS, and $t_{mn}(m, n \in \{i, j, k, l\})$ is the propagation delay from node m to node n . Assume $t_{mn} = t_{nm}$. Thus, we need to choose a T_{wait}^k satisfying

$$T_1 \leq T_{wait}^k \leq T_2 \quad (11)$$

where $T_1 = SIFS + 2(t_{ji} - t_{jk})$, and $T_2 = SIFS + t_{ji} + t_{il} - t_{jk} - t_{kl}$.

- **Overhearing a RTS:** Similarly, as shown in Fig. 3, if node k overhears RTS from node j , then, in order for node i and node l to be able to receive the DATA packet from node j and node k , respectively, we need

$$t_{ji} + 2 \cdot SIFS + CTS + t_{ij} + t_{ji} \leq t_{jk} + T_{wait}^k + t_{ki}$$

and

$$t_{jk} + T_{wait}^k + t_{kl} \leq t_{ji} + 2 \cdot SIFS + CTS + t_{ij} + t_{jl}$$

Thus, we need to choose a T_{wait}^k satisfying

$$T_3 \leq T_{wait}^k \leq T_4 \quad (12)$$

where $T_3 = 2 \cdot SIFS + CTS + 3t_{ij} - t_{jk} - t_{ki}$, and $T_4 = 2 \cdot SIFS + CTS + 2t_{ij} + t_{jl} - t_{jk} - t_{kl}$.

This solution has two limitations. First, it requires some position information, which results in higher cost. Second, it can only make concurrent transmissions happen after CTS is overheard and RTS is overheard if $T_1 \leq T_2$ in (11), and $T_3 \leq T_4$ in (12), respectively, i.e.,

$$t_{ij} + t_{kl} \leq t_{kj} + t_{il} \quad (13)$$

and

$$t_{ij} + t_{kl} \leq t_{ki} + t_{jl} \quad (14)$$

respectively, which means it can only address the problem in certain scenarios.

Solution Two: By carefully looking into this problem, we find that it is related to the *capture capability* of the nodes. Specifically, according to IEEE 802.11 standard, if a node first receives a packet, and then another packet arrives, the new packet can only be considered as interference, no matter how high the reception power of this new packet is. In other words, a node can only receive the packet that arrives first. However,

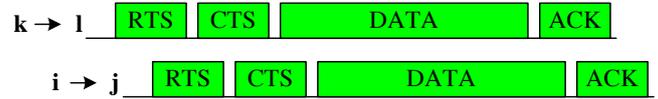


Fig. 6. Another example that two transmissions in Fig. 1 are partially overlapping.

if a node is able to turn to receive a new packet with much higher reception power when it has already begun to receive a packet, our problem shown in Fig. 5 can be solved.

Fortunately, there do exist such designs that support the capture of a stronger new packet. One example is Lucent's physical layer (PHY) design with "Message-In-A-Message" (MIM) support [3], where a MIM receiver is able to correctly detect and capture a strong frame during its reception of a weak frame [12].

Besides, with enhanced capture capability, some transmissions that cannot happen according to IEEE 802.11 can now be carried out. For example, in Fig. 1, if node k transmits RTS before node i does, node j will not be able to receive the RTS from node i according to IEEE 802.11 standard. However, with the enhanced capture capability discussed above, the transmission between node i and node j may be carried out as shown in Fig. 6.

In our proposed ATPMAC, we use the second solution as the default one while making the first solution as an option.

F. Comparison with POWMAC

The proposed ATPMAC seems similar to POWMAC [11] in the sense that in both protocols the nodes overhearing some transmissions are still allowed to carry out their own transmissions as long as they will not interfere with the ongoing ones. However, as we have shown above, ATPMAC has several advantages over POWMAC:

- Firstly, our proposed ATPMAC has much smaller overhead than POWMAC. In ATPMAC, $N(N \geq 1)$ concurrent DATA transmissions only need one RTS/CTS handshake. In contrast, according to POWMAC, N concurrent DATA transmissions should be preceded by N RTS/CTS handshakes.
- Secondly, POWMAC suffers from the problem shown in Fig. 5, which is caused by propagation delay, while ATPMAC has addressed that. Notice that this problem is one source of ineffectiveness in POWMAC because several transmissions may not be able to be carried out concurrently as planned.
- Thirdly, ATPMAC is much simpler than POWMAC. It is much easier to implement ATPMAC than to implement POWMAC, yet ATPMAC can achieve better performance than POWMAC as we will show in Section IV.

We notice that ATPMAC also has some limitations. For example, ATPMAC does not take node mobility into consideration, hidden problems still exist, and so on. We will investigate these issues in the future.

IV. PERFORMANCE EVALUATION

In this section, we use NS2 (version 2.29) to evaluate the proposed ATPMAC protocol and compare its performance

TABLE II
SIMULATION PARAMETERS.

Parameters	Value
Channel frequency	2.4 GHz
Data rate	1 Mbps
Basic rate	1 Mbps
SINR threshold	4 dB
Packet size	2000 bytes
Transmission range	250 meters
Carrier sensing range	250 meters
RTS retry limit	7

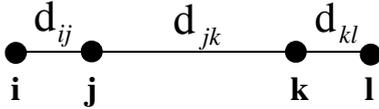


Fig. 7. A simple topology where every node is a one-hop neighbor of the other three nodes.

with POWMAC [11], and IEEE 802.11 MAC. We compare ATPMAC with POWMAC because the latter one is also a transmission power control MAC protocol based on a single-channel, single-transceiver design, and it shares some common features with ATPMAC. We do not compare ATPMAC with those protocols like [6] [7] because their main objective is to save energy and they can achieve comparable throughput to that of IEEE 802.11 MAC at best. Neither do we compare ATPMAC with those protocols with multi-channel and/or multi-transceivers [9] [10].

Besides, as we mentioned before, POWMAC suffers from the problem shown in Fig. 5 due to the existence of propagation delay. In order to make fair comparisons, we improve POWMAC by addressing this problem when we implement it. Some of our simulation parameters are shown in Table II.

A. A One-hop Scenario

We first conduct simulations in a one-hop scenario shown in Fig. 7, where the distances between node i and node j , between node j and node k , and between node k and node l , are denoted by d_{ij} , d_{jk} , and d_{kl} , respectively. There are two flows in this network, one from node i to node j , and the other from node k to node l .

Case 1: $d_{ij} = 35$ meters, $d_{jk} = 135$ meters, and $d_{kl} = 35$ meters. In this case, the two flows can happen concurrently without interfering with each other according to both ATPMAC and POWMAC, while only one flow can be carried out at a time according to IEEE 802.11 MAC.

The network throughput is shown in Fig. 8(a). Since the proposed ATPMAC does not introduce any additional overhead, the network throughput of ATPMAC is almost two times that of IEEE 802.11 MAC. POWMAC achieves about 82% higher throughput than IEEE 802.11 MAC due to the increased overhead. Thus, the network throughput of ATPMAC is higher than that of POWMAC by up to 10%.

Case 2: $d_{ij} = 90$ meters, $d_{jk} = 85$ meters, and $d_{kl} = 35$ meters. The network throughput is shown in Fig. 8(b). According to ATPMAC, two flows can be carried out concurrently only if node i and node j exchange RTS/CTS before node k and node l . Since about half of the time node i transmits RTS before node k and half of the time node k before node i , and thus, the network throughput of ATPMAC is about

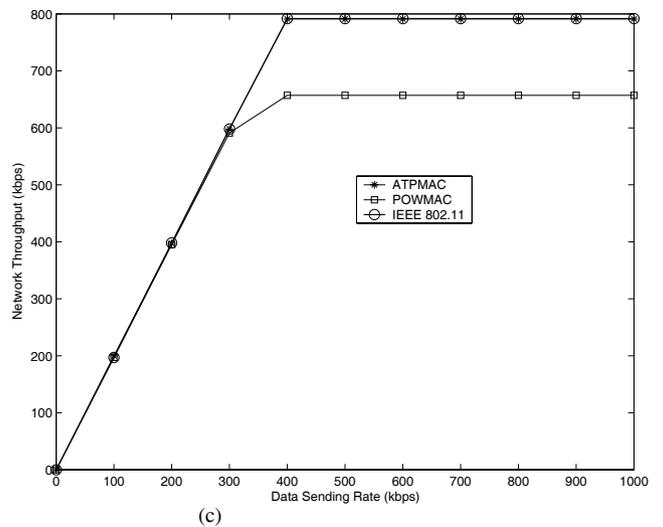
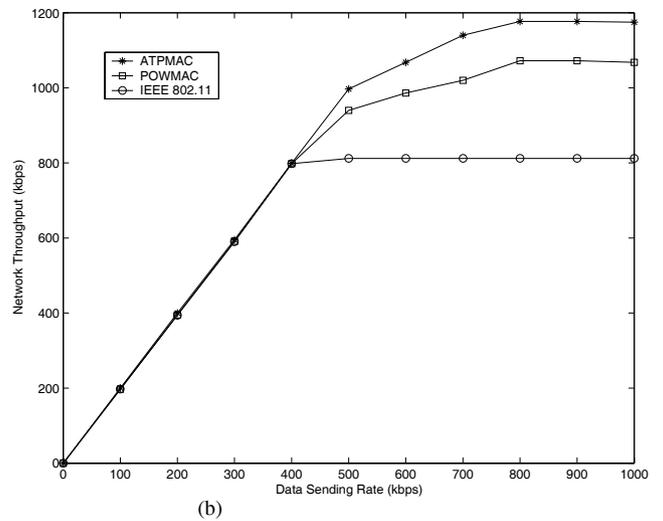
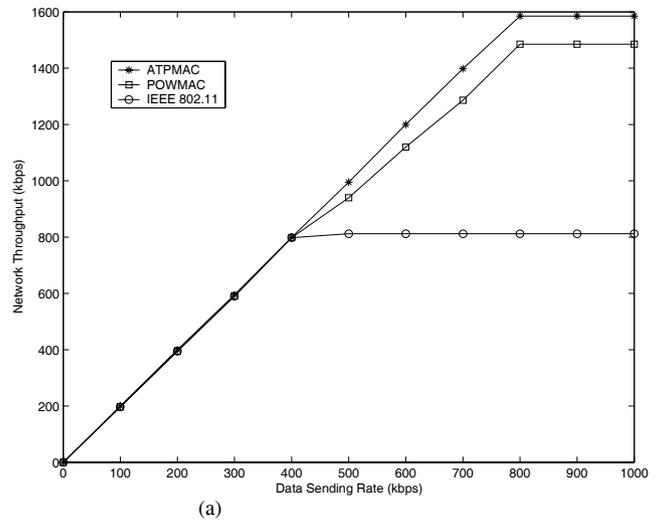


Fig. 8. Performance of ATPMAC, POWMAC, and IEEE 802.11 MAC (corresponding to the topology shown in Fig. 7). (a), (b), and (c) are the performance in Case 1, Case 2, and Case 3, respectively.

50% more than that of IEEE 802.11 MAC. POWMAC has a similar case in the sense that only half of the time can two transmissions happen concurrently. Because of additional overhead, the network throughput of POWMAC is about 32%

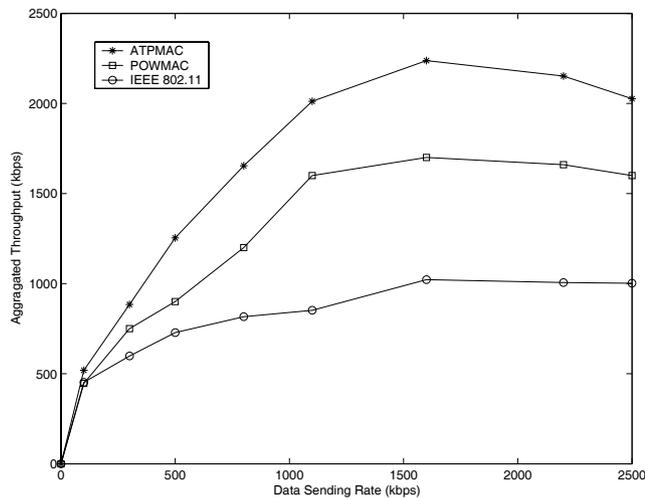


Fig. 9. The network throughput of ATPMAC, POWMAC, and IEEE 802.11 in a multi-hop scenario.

more than that of IEEE 802.11 MAC. Thus, the network throughput of ATPMAC is more than that of POWMAC by up to 13%.

Case 3: $d_{ij} = 135$ meters, $d_{jk} = 40$ meters, and $d_{kl} = 35$ meters. In this case, neither ATPMAC nor POWMAC can make two transmissions happen concurrently due to the serious interference between two flows. The network throughput is shown in Fig. 8(c). ATPMAC achieves almost the same throughput as that of IEEE 802.11 MAC, while POWMAC has degradation of about 20%. Thus, the network throughput of ATPMAC is more than that of POWMAC by up to 12%.

B. A Multi-hop Scenario

We then evaluate the performance of ATPMAC, POWMAC, and IEEE 802.11 MAC in a multi-hop scenario. We use a 1000m x 1000m 2D topology where there are 50 randomly distributed nodes. Ten nodes are chosen to be CBR (Constant Bit Rate) sources and their destination nodes are randomly chosen. The network uses AODV (Ad Hoc OnDemand Distance Vector Routing) routing protocol. The simulation parameters are the same as those shown in Table II except that the data rate is 2 Mbps and the SINR is 7db [1]. Fig. 9 shows the simulation result. We can see that ATPMAC can achieve up to 136% higher throughput than that of IEEE 802.11, and 32% higher throughput than that of POWMAC. This is because in multi-hop networks ATPMAC can make concurrent transmissions happen whenever possible without introducing any additional overhead, while POWMAC allows concurrent transmissions at the cost of more signaling overhead, i.e., N RTS/CTS exchanges for N concurrent transmissions. This overhead becomes more significant when the data rate increases.

As a remark, we must note that the actual performance of the original POWMAC should be worse than what we have shown above, because it does not address the imperfect

synchronization problem. In other words, the performance gain of ATPMAC over POWMAC is in fact larger than what we have shown.

V. CONCLUSION

In this paper, we propose a new adaptive transmission power controlled MAC protocol, known as ATPMAC, which can significantly improve the network throughput using a single channel and a single transceiver. Our simulations show that ATPMAC can improve the throughput by up to 136% compared to IEEE 802.11 MAC.

We must realize that there are some limitations on ATPMAC. First, it does not address the mobility issue. Second, hidden terminal problems still exist. We will investigate these issues in the future.

REFERENCES

- [1] <http://www.cisl.columbia.edu/grads/ozgun/ProjectApplication.html>.
- [2] Wireless LAN medium access control (MAC) and physical layer (PHY) specifications, IEEE Standards Working Group, 1999.
- [3] J. Boer, H. Bokhorst, W. Diepstraten, A. Kamerman, R. Mud, H. Driest, and R. Kopmeiners, Wireless LAN with enhanced capture provision, U.S. Patent 5987033, Nov. 1999.
- [4] P. Ding, J. Holliday, and A. Celik, "Demac: an adaptive power control MAC protocol for ad-hoc networks," in *Proc. IEEE 16th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'05)*, Berlin, Germany, Sept. 2005.
- [5] L. Jia, X. Liu, G. Noubir, and R. Rajaraman, "Transmission power control for ad hoc wireless networks: throughput, energy and fairness," in *Proc. IEEE Wireless and Communications and Networking Conference*, New Orleans, LA, USA, Mar. 2005.
- [6] E. Jung and N. Vaidya, "A power control MAC protocol for ad hoc networks," in *Proc. ACM MobiCom*, Atlanta, Georgia, USA, Sept. 2002.
- [7] P. Karn, "MACa: a new channel access method for packet radio," in *Proc. 9th ARRL Computer Networking Conference*, London, Ontario, Canada, 1990.
- [8] A. Lim and S. Yoshida, "A power adapted MAC (PMAC) scheme for energy saving in wireless ad hoc networks," *IEICE Trans. Fundamentals of Electronics, Commun. and Computer Sciences*, vol. E88-A, no. 7, pp. 1836–1844, July 2005.
- [9] J. Monks, V. Bharghavan, and W. Hwu, "A power controlled multiple access protocol for wireless packet networks," in *Proc. IEEE International Conference on Computer Communications (INFOCOM'01)*, Anchorage, Alaska, USA, Apr. 2001.
- [10] A. Muqattash and M. Krunz, "Power controlled dual channel (PCDC) medium access protocol for wireless ad hoc networks," in *Proc. IEEE International Conference on Computer Communications (INFOCOM'03)*, San Francisco, California, USA, Mar. 2003.
- [11] A. Muqattash and M. Krunz, "Powmac: a single-channel power-control protocol for throughput enhancement in wireless ad hoc networks," *IEEE J. Select. Areas Commun.*, vol. 23, no. 5, pp. 1067–1084, May 2005.
- [12] T. Nadeem, L. Ji, A. Agrawala, and J. Agre, "Location enhancement to IEEE 802.11 DCF," in *Proc. IEEE International Conference on Computer Communications (INFOCOM'05)*, Miami, FL, USA, Mar. 2005.
- [13] T. Rappaport, *Wireless Communications: Principles and Practice (Second Edition)*. Prentice-Hall PTR, 2002.
- [14] T. Rappaport and L. Milstein, "Effects of radio propagation path loss on DS-CDMA cellular frequency reuse efficiency for the reverse channel," *IEEE Trans. Veh. Technol.*, vol. 41, Aug. 1992.
- [15] H. Zhai and Y. Fang, "Physical carrier sensing and spatial reuse in multirate and multihop wireless ad hoc networks," in *Proc. IEEE International Conference on Computer Communications (INFOCOM'06)*, Barcelona, Spain, Apr. 2006.