

# A Jamming-Based MAC Protocol to Improve the Performance of Wireless Multihop Ad Hoc Networks

Shiang-Rung Ye, You-Chiun Wang, and Yu-Chee Tseng

**Abstract**—One critical issue in multihop ad hoc networks is the medium access control (MAC). The IEEE 802.11 MAC protocol is originally designed for fully connected, one-hop ad hoc networks but not for multihop ad hoc networks. In addition to the well known hidden-terminal problem, we found that IEEE 802.11 also suffers from an *erroneous reservation problem* which occurs when RTS-CTS exchange fails but the channel is incorrectly reserved. In this paper, we propose a *jamming-based MAC (JMAC)* protocol that is not only free from both the hidden-terminal and the erroneous reservation problems but also allows more concurrent transmission/receipt activities for stations within each other's transmission range. The idea behind the JMAC is to separate source stations' traffic from destination stations' traffic into different channels (i.e., dividing the shared medium into two channels), and explicitly signal the channel status by jamming the channels. Simulation results show that although the channel division incurs some cost, the advantages of being free from the erroneous reservation and the hidden terminal problems, and the benefits of more concurrent transmissions will compensate the cost and provide higher channel utilization when data frame size is median or large.

**Index Terms**—medium access control (MAC), mobile ad hoc network, wireless communication.

## 1 INTRODUCTION

An *ad hoc network* is a spontaneous network that consists solely of mobile stations without base stations. It can be applied in many contexts such as military communication, disaster rescue, and outdoor/indoor activities due to its features of convenience in deployment and flexibility in reconfiguration. Depending on configurations, an ad hoc network can be classified as *fully connected* or *multihop*. In this work, we focus on multihop ad hoc networks.

One of the issues in multihop ad hoc networks is the medium access control (MAC). Traditional MAC protocols such as ALOHA [1] and CSMA [2] all suffer from the well-known hidden-terminal problem. The hidden-terminal problem occurs when some stations are hidden from the source stations but transmit while their neighbors are receiving data. In the literature, many methods have been proposed to solve the problem [3]–[12]. The BTMA [3] first proposed using busy tone to avoid the hidden-terminal problem in an infrastructure network. In the BTMA, the whole bandwidth is divided into a data channel and a busy-tone channel. When a base station detects carrier on the data channel, it transmits busy-tone signal (a sine wave) to indicate the busy state of the data channel. Any mobile station that detects the busy-tone signal will defer its transmission until the end of the busy tone. The disadvantages of using busy tone are the cost of bandwidth for the busy tone and the cost of time to detect busy-tone signal which is no longer negligible on a narrow-band channel and must be accounted for [3].

In the receiver-initiated busy-tone multiple protocol [4], a receiver will transmit a busy tone after it receives a request frame from a sender. This busy tone not only serves as a response to the request but also prevents other stations from

accessing the channel since any attempting source that detects the busy tone will postpone their transmissions. However, in a multihop ad hoc network, a sender may overhear neighboring stations' busy tone, and it is difficult to tell whether it is its receiver or other stations turning on the busy tone.

Based on the RTS-CTS-DATA dialogue, the DBTMA protocol [10] [11] uses two busy tones to convey channel states. It divides the communication bandwidth into a control channel, a data channel, and two busy tones ( $BT_t$  and  $BT_r$ ). RTS and CTS frames are exchanged on the control channel, and if the exchange succeeds, a source station will transmit its data on the data channel and turn on the  $BT_t$  busy tone to prevent neighboring stations from interfering its communication. Similarly, while a destination is waiting for or receiving data, it turns on the  $BT_r$  to avoid possible collision.

Another branch of MAC design is based on the RTS-CTS exchange. The MACA protocol [5] first introduces the RTS-CTS exchange to prevent the hidden-terminal problem. In this scheme, a source station transmits a RTS frame to destination station for request of transmission. If the intended destination correctly receives the RTS frame, it will admit the transmission by sending a CTS frame. Other stations that hear the RTS or CTS frame are required to reschedule their transmissions at later time for preventing frame collision. In contrast to MACA, the MACA-BI [7] which is a receiver-oriented protocol tries to improve the channel utilization by removing the RTS part of RTS/CTS handshake. A destination station sends a RTR (ready-to-receive) frame to invite a source station to transmit data. The RTR frames are transmitted at a rate that matches the source station's incoming traffic according to backlog information of the source station, which is piggybacked in the source's data frames, for helping the destination to predict the source station's traffic. However, this method is not suitable for networks with unpredictable traffic such as burstiness traffic.

The MACAW [5] protocol suggests a new frame exchange

The authors are with the Department of Computer Science and Information Engineering, National Chiao-Tung University, Hsin-Chu, 30010, Taiwan.  
E-mail: {shiarung, wangyc, yctsen}@csie.nctu.edu.tw

of RTS-CTS-DS-DATA-ACK. A DS (Data-Sending) frame is transmitted by a source station to confirm the use of the medium after it receives a CTS frame. However, since a DS frame may collide with frames of other stations, neighboring stations may not correctly receive the DS frame. The ACK frame is transmitted after a receiver correctly receives a data frame. This acknowledgement scheme improves the reliability of a wireless link, and avoids the long recovery cost at the upper layer (e.g., TCP).

The FAMA [8] [9] suggests that a station must acquire a channel before transmission of data. One way to acquire the channel is through the RTS-CTS exchange. Two variants of FAMA (FAMA-NCS and FAMA-NPS) are discussed and they are similar to the MACA and IEEE 802.11 [12], respectively.

The IEEE 802.11 standard adopts the RTS-CTS-DATA-ACK exchange sequence when the size of a data frame is larger than a threshold. In fact, in a large-scale network or in situations where the hidden-terminal problem is unavoidable, this exchange sequence is suggested to be used in most cases [13]. As in the MACAW, the RTS and CTS may collide in IEEE 802.11 so the hidden-terminal problem remains unavoidable (but will be reduced). Besides, in this work, we observe that IEEE 802.11 also suffers from an *erroneous reservation* problem which occurs when a RTS/CTS exchange fails but the medium has been incorrectly reserved by the RTS and/or CTS frames. During this reserved period, all stations in the reserved area are suppressed from transmission even though the channel is idle.

In this paper, we propose a *jamming-based MAC (JMAC)* protocol that can satisfactorily solve the erroneous reservation problem and the hidden terminal problem. In addition, the proposed JMAC protocol also allows more concurrent transmission pairs in the physical area, thus further increasing the channel utilization. The basic idea behind the JMAC is to separate source stations' traffic from destination stations' traffic by dividing the shared medium into two different channels, and explicitly signal the channel status by jamming the channels. Although the division of the shared medium into two channels incurs some cost, as shown in simulation results, the advantages of being free from the erroneous reservation and the hidden terminal problems, and the benefits of more concurrent transmissions will compensate the cost and provide higher channel utilization when data frame size is median or large.

The rest of this paper is organized as follows: In Section 2, we review the IEEE 802.11 standard and some deficiencies of IEEE 802.11 when it is applied to a multihop ad hoc network. In Section 3, we describe the operations of the proposed JMAC protocol. Simulation results are shown in Section 4, and conclusions are drawn in Section 5.

## 2 DEFICIENCIES OF IEEE 802.11 IN MULTIHOP AD HOC NETWORKS

The basic access method of IEEE 802.11 is the distributed coordination function (DCF). It specifies how to determine channel states and how to backoff when transmission fails. Two kinds of carrier senses are defined in the DCF: *physical carrier sense* and *virtual carrier sense*. The former is supported by the physical layer, while the latter is conducted by the MAC layer. The virtual carrier sense is carried out by the *network allocation vector (NAV)*, which is set according to the *ID/Duration* field of data/control frames. The *ID/Duration* field contains

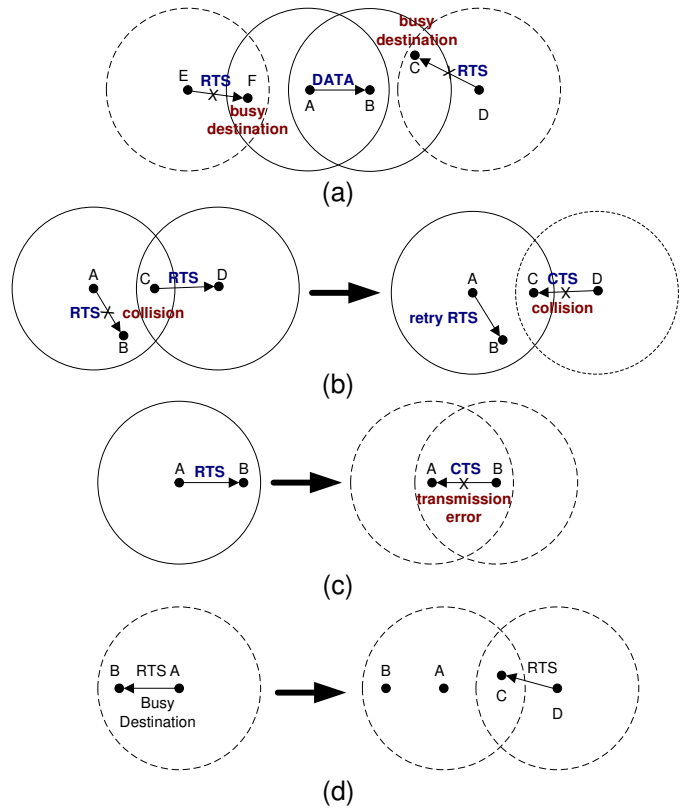


Fig. 1: Erroneous reservations caused by (a) busy destination, (b) frame collision, (c) transmission error, and (d) the problem itself.

the information of a time period during which some stations will use the medium. By this information, stations are able to know the future use of the medium and schedule their transmissions to avoid collision.

IEEE 802.11 also adopts the RTS-CTS-DATA-ACK as one of frame exchange sequences. When the size of a data frame is larger than 3000 bits (a default value) [12], this exchange sequence is recommended to be used. The functions of RTS and CTS frames are twofold. First, they are used to probe the channel state so as to prevent collision of large data frames. Second, they contain NAV information to prevent the hidden terminal problem. However, since RTS/CTS frames may collide with other frames, the hidden terminal may still occur.

Besides, the use of RTS/CTS in 802.11 also causes an erroneous reservation problem, which may occur when RTS/CTS exchange fails but the channel is reserved by the RTS/CTS frames. This inhibits neighboring stations from accessing the medium even though the medium is idle. Such *holding-while-waiting* scenario may waste much bandwidth when the traffic load is high. The erroneous reservation may occur under the following conditions: busy destination, frame collision, transmission error, and the problem itself. For example, in Fig. 1(a), assume that stations A and B have successfully completed RTS/CTS exchange and started their transmission. In the meanwhile, if station D transmits a RTS to C, the circle centered at D will be incorrectly reserved. Similarly, if E sends a RTS to F, the circle centered at E will be incorrectly reserved too.

Frame collision may also cause erroneous reservations. Fig. 1(b) shows that if the CTS\_Timeout interval is smaller than the length of a CTS frame, the erroneous reservation may occur. In the example, A and C transmit RTS frames at the

same time. The RTS frame from  $A$  is collided with that from  $C$ , but the RTS frame from  $C$  is successfully transmitted to  $D$ . While  $D$  responds with a CTS frame to  $C$ , if  $A$  retries to send a RTS frame, the CTS frame is collided at  $C$  and the circle centered at  $D$  is erroneously reserved.

Fig. 1(c) demonstrates that transmission errors can also cause erroneous reservations. In the figure,  $B$  successfully receives the RTS from  $A$ , but  $A$  fails to receive the CTS from  $B$  due to transmission error. Then the circles centered at both  $A$  and  $B$  will be incorrectly reserved. It is also possible that after a station incorrectly reserves the channel, this incorrect reservation causes another reservation. In Fig. 1(d), after  $A$  incorrectly reserves the channel,  $D$  also incorrectly reserves the channel.

Note that the duration of an erroneous reservation may vary depending on different cases. If a RTS is received, a station will set its NAV according to the corresponding frame length. However, if the station fails to hear any frame after a timeout period, the NAV will be reset and the length of erroneous reservation interval would be shorten. The examples in Fig. 1(a) and (d) fall into this case. For a station hearing a CTS, the NAV can not be reset. In this case, the erroneous reservation interval would be much longer. The examples in Fig. 1(b) and (c) fall into this case.

### 3 THE PROPOSED MAC PROTOCOL

In this section, we describe the proposed MAC protocol, called *JMAC*, that is derived based on the concept of traffic separation and jamming mechanism. In *JMAC*, the medium is divided into two channels:  $S$  channel and  $R$  channel. RTS and DATA frames (source stations' traffic) are transmitted on the  $S$  channel and CTS and ACK frames (destination stations's traffic) are transmitted on the  $R$  channel. It is assumed that each station is equipped with two radio devices, one tuned to the  $S$  channel and the other tuned to the  $R$  channel. The ratio of bandwidth allocated to the  $R$  and  $S$  channels is assumed to be  $\alpha : (1 - \alpha)$ , where  $0 < \alpha < 1$ . How to choose an appropriate  $\alpha$  will be further discussed in Section 3.2.

#### 3.1 Protocol Behaviors

In *JMAC*, a source station always transmits RTS/DATA frames on the  $S$  channel but receives CTS/ACK frames on the  $R$  channel. It also transmits jamming signal on the  $S$  channel while waiting or receiving a CTS/ACK frame on the  $R$  channel. For a destination station, while it is waiting or receiving a DATA frame on  $S$  channel, it jams the  $R$  channel to prevent neighboring stations from transmitting RTS frames on the  $S$  channel. Jamming signal is the one with sufficient energy causing the medium to become busy. No data is carried in jamming signal and there is no need to decode its content. The overlapping of a jamming signal and a data signal is considered as a jamming signal too, and thus can not be correctly recognized.

The procedures and the timing diagram of transmission and receipt of RTS, CTS, DATA, and ACK frames in *JMAC* are shown in Figs. 2 and 3, respectively. Before transmitting a RTS frame on the  $S$  channel, a source station first senses the  $R$  channel. If it is idle for a DIFS period, the source station sends a RTS on the  $S$  channel, and then listens to the  $R$  channel for a CTS frame. If a station detect  $R$  channel to be busy before sending a RTS frame, this implies that some neighbors may be receiving data from other stations that are two hops way so

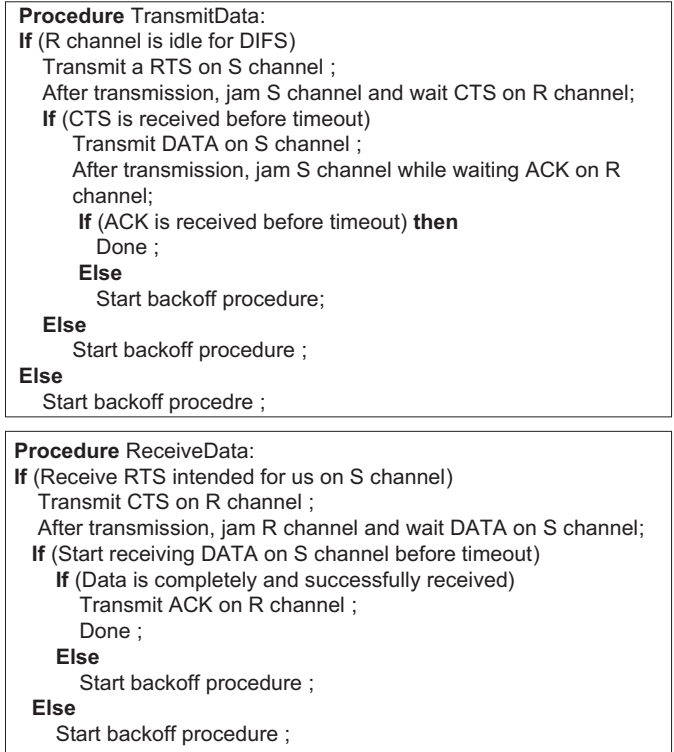


Fig. 2: The procedures of transmission and receipt of control/data frames in *JMAC*.

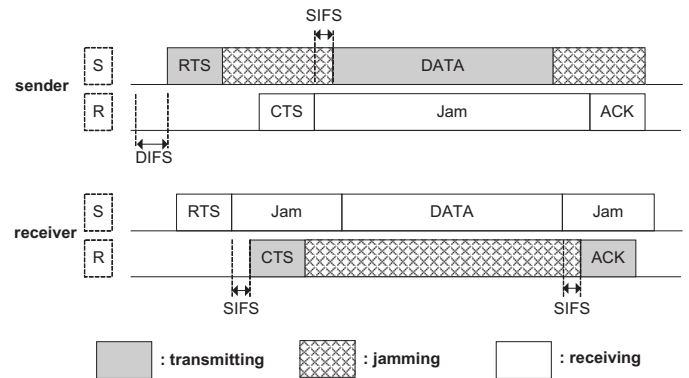


Fig. 3: The timing diagram of transmission of control/data frames.

the source station is not allowed to access the medium. While waiting for the CTS frame from a intended destination, the source station is required to jam the  $S$  channel. The purpose of jamming the  $S$  channel is similar to the reservation function of RTS in 802.11, but the difference is that the medium is jammed as long as needed, depending on the result of RTS-CTS exchange. If the RTS-CTS exchange fails (indicated by a CTS\_Timeout), the sender will stop jamming the  $S$  channel and will start the backoff procedure as in 802.11.

After the destination station receives the RTS from the  $S$  channel, it responds with a CTS frame on the  $R$  channel, and then listens to the  $S$  channel for a DATA frame. While it is waiting for the DATA frame, it also jams the  $R$  channel to prevent neighbors from transmitting RTS frames. If unfortunately, the DATA frame fails to appear after timeout, it will stop jamming the  $R$  channel.

The rest of access procedure is similar. After receiving the CTS, the source transmits its DATA frame, and then jams the  $S$  channel while waiting for an ACK on the  $R$  channel. For the

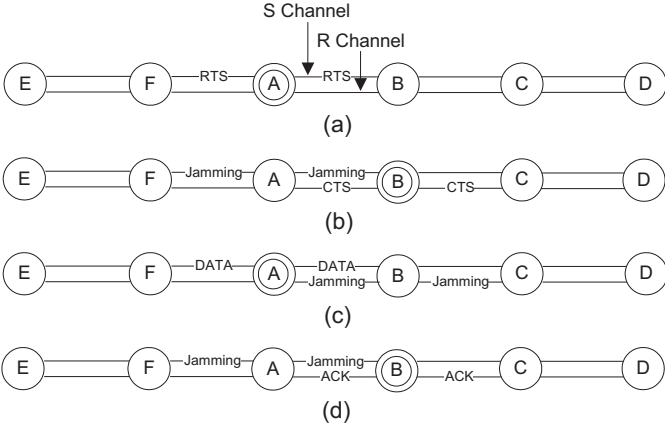


Fig. 4: An example of frame exchange and the transmission of jamming signal in the JMAC protocol.

destination, after receiving the data frame successfully, it will respond with an ACK on the  $R$  channel. Note that although we adopt the RTS-CTS-DATA-ACK frame exchange sequence, stations may also apply the DATA-ACK sequence directly to transmit data frames. The risk is a higher penalty in case that data frames from two sources collide.

In JMAC, the backoff procedure starts after the  $R$  channel becomes idle for one DIFS period, and it is independent of the status of the  $S$  channel. This is because the transmission of a RTS on the  $S$  channel doesn't interfere with neighboring stations' reception of CTS/ACK frames on the  $R$  channel, but the reception of RTS/DATA frames on the  $S$  channel.

Fig. 4 illustrates the exchange of RTS, CTS, DATA, and ACK frames in a network that stations are arranged in a line. The double circles, in the example, represent the stations that are currently transmitting RTS, CTS, DATA, or ACK frames. Fig. 4 (a) and Fig. 4 (b) show that  $A$  is transmitting a RTS frame to  $B$  and then jamming the  $S$  channel, respectively. Since the RTS frame is transmitted by broadcasting,  $F$  also hears the RTS frame. After receiving the RTS frame,  $B$  responds with a CTS frame and then jams the  $R$  channel in Fig. 4(b) and Fig. 4(c) respectively. This will prevent the hidden terminal problem and erroneous reservation problem. Assume that if  $C$  misses the CTS frame from  $B$  due to collision or transmission error,  $C$  will not be a hidden terminal since it will detect the busy  $R$  channel and know that some of neighboring stations are receiving data. Also, since  $B$  will stop jamming the  $R$  channel if it doesn't receive a DATA frame after timeout, the erroneous reservation problem will not occur in JMAC. After  $A$  receives the CTS frame from  $B$ , it transmits a DATA frame and jams the  $S$  channel in Fig. 4(d). It can be easily observed that the jamming signal transmitted by  $A$  also protects its receipt of the ACK frame from  $B$ . We comment that if  $F$  doesn't successfully receive the RTS frame from  $A$  in Fig. 4(a), since  $A$  will transmit RTS/DATA frames and jamming signal on the  $S$  channel during the time period of RTS-CTS-DATA-ACK exchange, any RTS frame sent to  $F$  will be collided at  $F$ . Therefore,  $F$  will not transmit any CTS or ACK frame during this time period.

JMAC allows more concurrent transmission/receipt activities for stations within each other's transmission range. Taking Fig. 4(b) as an example, JMAC allows  $D$  and  $F$  to concurrently request transmission to  $C$  and  $E$ , respectively. After  $C$  and  $E$  receive RTS frames from  $E$  and  $F$ , they can safely reply with CTS frames without collision. This is shown in Fig. 5(a) and

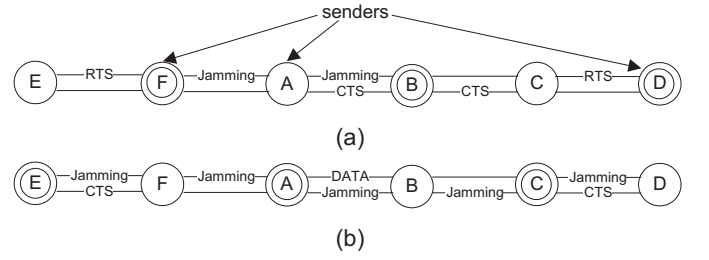


Fig. 5: An example of concurrent transmissions in the JMAC protocol.

Fig. 5(b).

### 3.2 Tuning the Factor $\alpha$

In this section, we discuss how to choose the ratio  $\alpha$  that determines the bandwidths of the  $S$  channel and  $R$  channel. Let the system transmission rate be  $r$ . After dividing the total bandwidth into two sub-channels, the transmission rates for the  $S$  channel and  $R$  channel are assumed to be  $\alpha \times r$  and  $(1 - \alpha) \times r$ , respectively. The basic idea is to find a value of  $\alpha$  such that the time of a RTS-CTS-DATA-ACK exchange, denoted by  $f(\alpha)$ , is minimized. The  $f(\alpha)$  can be expressed as follows:

$$f(\alpha) = \frac{RTS + DATA}{\alpha \times r} + \frac{CTS + ACK}{(1 - \alpha) \times r} \quad (1)$$

Differentiating  $f(\alpha)$  with respect to  $\alpha$ , we have

$$\frac{df(\alpha)}{d\alpha} = \frac{\alpha^2 \times (CTS + ACK) - (1 - \alpha)^2 \times (RTS + DATA)}{\alpha^2 \times (1 - \alpha)^2 \times r} \quad (2)$$

Set Eq. (2) to zero, we have the optimal value  $\hat{\alpha}$ ,

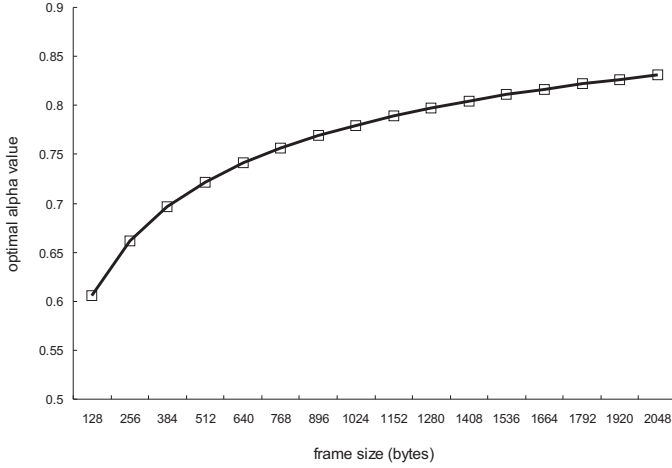
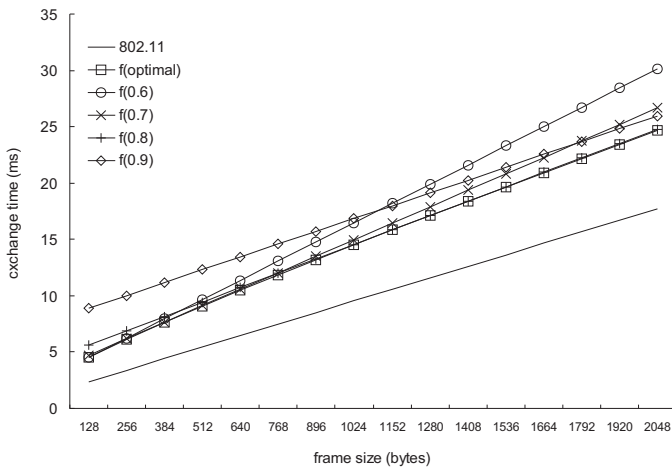
$$\hat{\alpha} = \frac{RTS + DATA \pm \sqrt{(RTS + DATA) \times (CTS + ACK)}}{RTS + DATA - CTS - ACK} \quad (3)$$

Since  $0 < \hat{\alpha} < 1$ ,

$$\hat{\alpha} = \frac{RTS + DATA - \sqrt{(RTS + DATA) \times (CTS + ACK)}}{RTS + DATA - CTS - ACK} \quad (4)$$

Although the sizes of RTS, CTS, and ACK frames are all fixed, the size of data frames may vary. Fig. 6 shows the relation of the optimum values of  $\hat{\alpha}$  and the sizes of data frames,  $s$ . The optimal  $\hat{\alpha}$  tends to increase as the frame size increases. In other words, a larger data frame would require more bandwidth to be assigned to the  $S$  channel. The figure also shows that there is no globally optimal  $\hat{\alpha}$  for all frame sizes. So we turn to search a approximation to the global optimal for the range of frame sizes specified in 802.11. In Fig. 7, we plot the frame exchange time of IEEE 802.11 and JMAC for different data frame sizes and different  $\alpha$  values. Note that for JMAC, the curve of  $f(\text{optimal})$  represents the ideal case where the data frame size is always known and we can always choose the best  $\hat{\alpha}$  to minimize the frame exchange time. As can be seen in the figure,  $\alpha = 0.7-0.8$  is a good approximation to the curve of  $f(\text{optimal})$  when the frame size falls in the range of 128–2048 bytes.

The frame-exchange time of RTS-CTS-DATA-ACK in 802.11 is also shown in Fig. 7. Since 802.11 fully utilizes channel bandwidth, its frame-exchange time is shorter than that of JMAC. This reflects the cost incurred by JMAC due to channel division. However, the above analysis is under the ideal


Fig. 6: Data frame size vs. the optimum alpha  $\hat{\alpha}$ .

Fig. 7: Frame exchange time in IEEE 802.11 and in JMAC for different data frame sizes and different  $\alpha$ .

assumption that IEEE 802.11 always successfully completes its frame exchange. As discussed earlier, IEEE 802.11 may suffer from the hidden terminal and erroneous reservation problems. Thus, such an advantage may be offset by these factors.

#### 4 PERFORMANCE EVALUATION

In this section, we present our simulation results and compare the performance of JMAC to that of IEEE 802.11. Parameters in Table 1, Table 2, and Table 3 are used in the simulation. In the simulation,  $N$  stations are uniformly placed in a  $120\text{m} \times 120\text{m}$  area. The transmission range of each station is  $R = 30$  m and the transmission rate is 1 Mbps. In each individual simulation run, the data frame size is assumed to be fixed, and the movement of a station follows a two-state model in which each station transits from the *moving* state to the *still* state with probability  $P_S$  and from the *still* state to the *moving* state with probability  $P_M$ . When transiting from the *still* to the *moving* state, a station chooses one of eight directions and moves in that direction with a constant speed of 1 m/sec. The total simulation time is 50 minutes in each simulation run, and frames are assumed to arrive at each station according to the Poisson process. As to the bandwidth of the S channel and R channel, we choose  $\alpha = 0.78$ , which is the optimal  $\alpha$  for the frame size of 1024 bytes.

Figs. 8 and 9 show the aggregate throughput and mean throughput of JMAC and IEEE 802.11 under different traffic

parameters	values
transmission rate ( $r$ )	1 (Mb/sec)
RTS	20 bytes
CTS	14 bytes
ACK	14 bytes
DATA	1024 bytes
number of retry	7
$CW_{\min}$	31
$CW_{\max}$	1023
CTS_Timeout	$SIFS + 2 \times \tau$
DATA_Timeout	$SIFS + 2 \times \tau$
ACK_Timeout	$SIFS + 2 \times \tau$
duration/ID (for RTS)	$CTS + DATA + ACK + 3 \times SIFS + 3 \times \tau$
duration/ID (for CTS)	$DATA + ACK + 2 \times SIFS + \tau$

TABLE 1: MAC Layer parameters.

parameters	values
AslotTime	20 $\mu\text{s}$
ASIFSTime	10 $\mu\text{s}$
ADIFSTime	50 $\mu\text{s}$
PLCP preamble	24 bytes
PLCP header	6 bytes

TABLE 2: Physical layer parameters.

parameters	values
bandwidth ratio ( $\alpha$ )	0.784
propagation delay ( $\tau$ )	1 $\mu\text{s}$
transmission range ( $R$ )	30 m
simulated area ( $A$ )	$4R \times 4R \text{ m}^2$
moving speed ( $v$ )	1 m/s
still probability ( $P_S$ )	0.1
moving probability ( $P_M$ )	0.9

TABLE 3: Parameters of the simulated wireless network.

loads. At light loads (form 1 to 5 frames/sec), the mean throughputs of JMAC and IEEE 802.11 are very close. However, as the traffic load increases, unless for small  $N$  (such as  $N = 20$ ), the throughputs of JMAC will outperform the throughputs of IEEE 802.11. This implies that at this stage, the hidden terminal and erroneous reservation problems start to degrade the performances of IEEE 802.11. On the contrary, JMAC is quite resistant to such effects, and thus can still perform very well. As classic multiple access protocols such as ALOHA and CSMA, IEEE 802.11 also exhibits instability on channel throughput. Therefore, after reaching the overload condition, the performance of 802.11 start to degrade. For the case of  $N = 20$ , since the traffic load is still light, the network is not saturated yet and the throughputs of JMAC and 802.11 are quite the same. But as traffic load increases, shown in Fig. 10, the similar behavior can also be observed.

Figs. 10 and 11 further demonstrate the saturation throughputs under different network sizes  $N$ . Although, saturation throughput decrease with the increase of  $N$  in both JMAC and 802.11, JMAC still shows much better performance than 802.11. The saturation throughput may be affected by many factors: the number of stations in the network, the maximum backoff window, retry count, and the hidden terminal problem [13], [14]. In this simulation, the same parameters are used for both JMAC and 802.11, which may imply that the hidden terminal problem and the erroneous reservation problem are two causes of degradation of the performance of 802.11. This also justifies value of the central design of our JMAC protocol by separating traffics into two channels and explicitly signaling channel status.

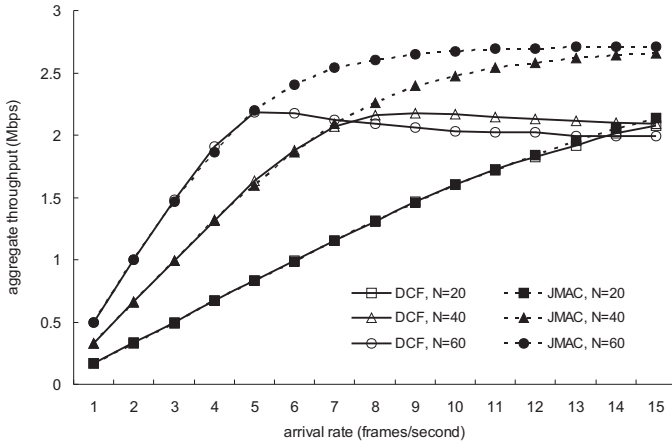


Fig. 8: Aggregate throughput versus frame arrival rate.

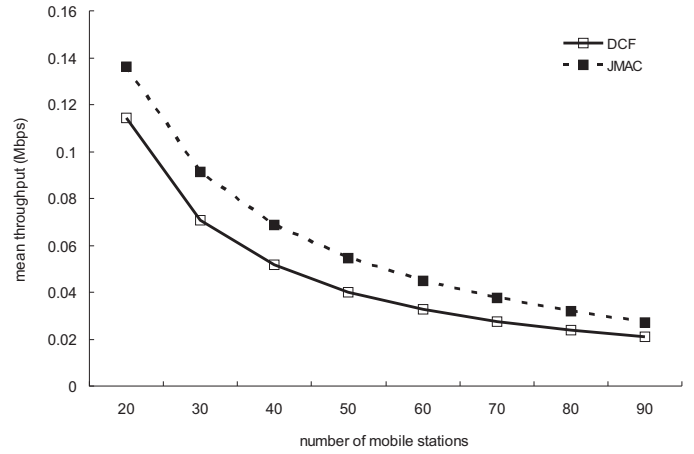


Fig. 11: Saturation mean throughput versus network density.

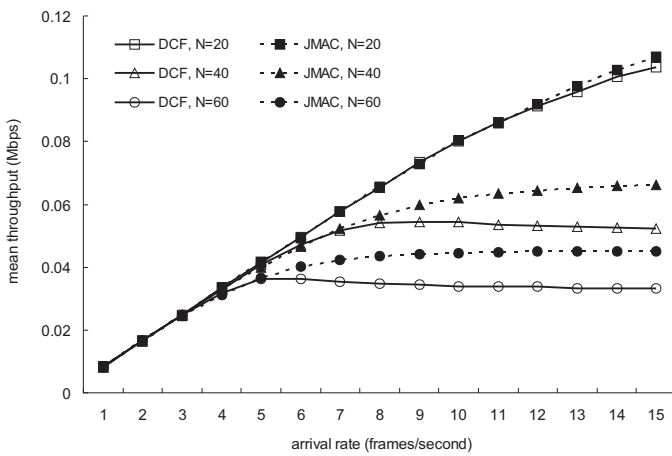


Fig. 9: Mean throughput versus frame arrival rate.

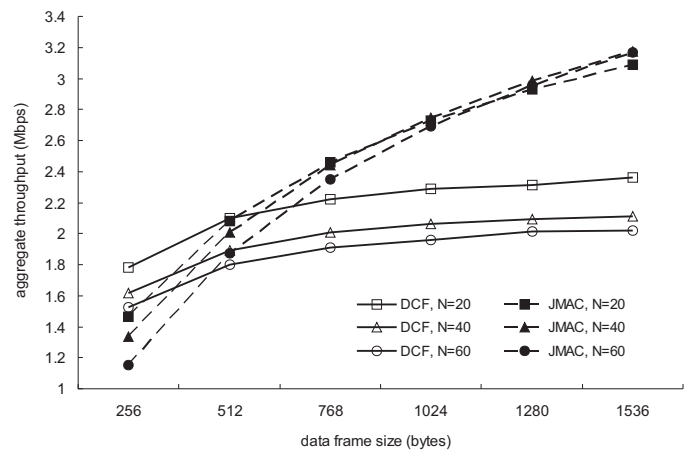


Fig. 12: Aggregate throughput versus frame size.

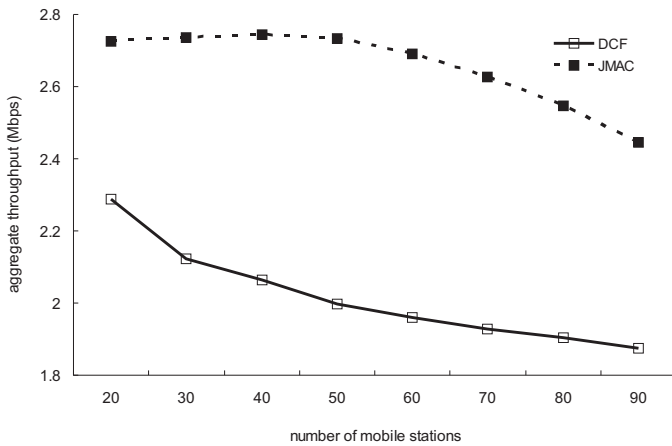


Fig. 10: Saturation aggregate throughput versus network density.

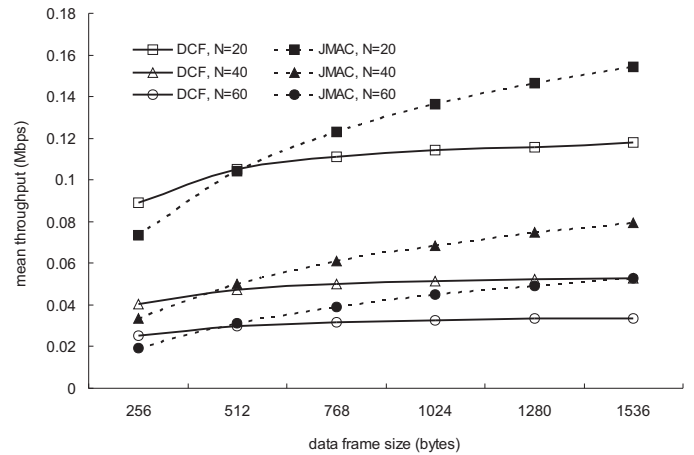


Fig. 13: Mean throughput versus frame size.

In the above simulations, the data frame size is fixed at 1,024 bytes. In this part, we vary the data frame size to observe its effect. As Figs. 12 and 13 show, both JMAC and 802.11 benefit from larger data frames due to less control overheads incurred by RTS, CTS, and ACK frames. The results also show that smaller frames will favor 802.11, but larger frames will favor JMAC. For 802.11, the impact of the hidden-terminal problem will become more serious for larger data frames (due to higher penalties caused by collisions). On the contrary, JMAC doesn't suffer from such a problem and can thus benefit

from this factor. Hence, JMAC is more efficient than IEEE 802.11 for transmitting medium or large data frames (> 512 bytes/frame).

Fig. 14 presents mean access delay of data frames versus frame arrival rate. The access delay increases with frame arrival rate. Under low traffic load, the access delay of JMAC is longer than that of 802.11. This is because the time to complete the frame exchange sequence is longer than that in 802.11. But as traffic load increases, 802.11 will suffer from more collision, and thus its access delay dramatically increases.

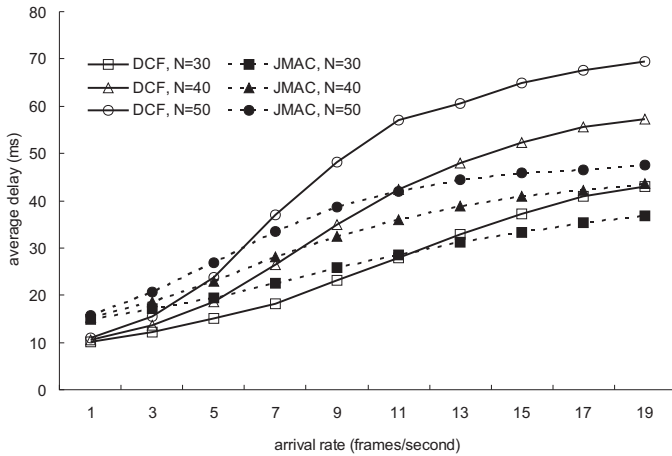


Fig. 14: Mean access delay versus frame arrival rate.

## 5 CONCLUSIONS

Besides the well known hidden terminal problem, the IEEE 802.11 also suffers from the erroneous reservation problem. In this paper, we propose a jamming-based MAC (JMAC) protocol which is free from both the hidden terminal and erroneous reservation problems, and thus provides higher channel utilization than IEEE 802.11 for medium or large frame sizes. JMAC separates source stations' traffic from destination stations' traffic into different channels, and explicitly signals the channel status by jamming the channels. We also discuss how to choose a proper ratio for determining the bandwidths of the channels. It is shown that, the optimal ratio changes with the data frame size so there is no global optimal value for all sizes. But the values in the range of 0.7 and 0.8 would be a good approximation for frame size ranging from 128 to 2048 bytes. It is also shown that channel division incurs some cost in terms of the transmission time of a RTS-CTS-DAT-ACK exchange sequence; however, from simulation results, the advantages of being free from the erroneous reservation and the hidden terminal problems and the benefit of more concurrent transmissions of JMAC can compensate the cost of channel division and provide higher throughput than IEEE 802.11.

## REFERENCES

- [1] N. Abramson, "The aloha system - another alternative for computer communication," *Proc. AFIPS Computer Conf.*, pp. 281-285, 1970.
- [2] F.A. Tobagi and L. Kleinrock, "Packet switching in radio channels: part I - carrier sense multiple-access modes and their throughput-delay characteristics," *IEEE Trans. Comm.*, vol. 23, no. 12, 1975.
- [3] 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. Comm.*, vol. 23, no. 12, 1975.
- [4] C. Wu and V. Li, "Receiver-initiated busy-tone multiple access in packet radio networks," *Proc. ACM Workshop Frontiers in Computer Comm. Technology*, pp. 336-342, 1988.
- [5] P. Karn, "MACA: a new channel access method for packet radio," *Proc. Computer Networking Conf.*, 1990.
- [6] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: a media access protocol for wireless LAN's," *Proc. ACM Conf. Comm. Architectures, Protocols and Applications*, pp. 212-225, 1994.
- [7] F. Talucci, M. Gerla, and L. Fratta, "MACABI (MACA by invitation): a receiver oriented access protocol for wireless multiple networks," *Proc. IEEE Int'l Symp. Personal, Indoor and Mobile Radio Comm.*, pp. 435-439, 1997.

- [8] C.L. Fullmer and J.J. Garcia-Luna-Aceves, "Floor acquisition multiple access (FAMA) for packet-radio networks," *Proc. ACM Conf. Applications, Technologies, Architectures, and Protocols for Computer Comm.*, pp. 262-273, 1995.
- [9] C. Fullmer and J.J. Garcia-Luna-Aceves, "Solutions to hidden terminal problems in wireless networks," *Proc. ACM Conf. Applications, Technologies, Architectures, and Protocols for Computer Comm.*, pp. 39-49, 1997.
- [10] J. Deng and Z.J. Haas, "Dual busy tone multiple access (DBTMA): a new medium access control for packet radio networks," *Proc. IEEE Int'l Conf. Universal Personal Comm.*, pp. 973-977, 1998.
- [11] Z.J. Haas and J. Deng, "Dual busy tone multiple access (DBTMA): Performance evaluation," *Proc. IEEE Vehicular Technology Conf.*, pp. 314-319, 1999.
- [12] *IEEE Standard 802.11-1999, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications*, 1999, IEEE.
- [13] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE J. Selected Areas in Comm.*, vol. 18, no. 3, pp. 535-547, 2000.
- [14] G. Bianchi, L. Fratta, and M. Oliveri, "Performance analysis of IEEE 802.11 CSMA/CA medium access control protocol," *Proc. IEEE Int'l Symp. Personal, Indoor and Mobile Radio Comm.*, pp. 407-411, 1996.