Design of Multichannel MAC Protocols for Wireless Ad-Hoc Networks

Shou-Chih Lo

Department of Computer Science and Information Engineering

National Dong Hwa University, Hualien 974, Taiwan

Email: sclo@mail.ndhu.edu.tw

Tel: +886-3-8634029, Fax: +886-3-8634010

Medium access control (MAC) protocols coordinate channel access between wireless stations, and they significantly affect the network throughput of wireless ad-hoc networks. MAC protocols that are based on a multichannel model can increase the throughput by enabling more simultaneous transmission pairs in the network. In this paper, we comprehensively compare different design methods for multichannel MAC protocols. We classify existing protocols into different categories according to the channel negotiation strategies they employ. The common problems that may be encountered in multichannel design are discussed. We then propose a hybrid protocol that combines the advantages of the two methods of a common control channel and a common control period. The simulation results show that our proposed protocol can significantly outperform two representative protocols.

Keywords: Wireless Ad-Hoc Networks; IEEE 802.11; Medium Access Control; Multichannel Protocol

1. INTRODUCTION

Wireless local area networks (WLANs) are becoming increasingly popular, and can be configured as infrastructure (comprising an access point and the associated wireless stations [STAs]) or ad-hoc (comprising STAs only) networks. The wireless ad-hoc network is a self-organizing and self-configuring multihop wireless network, and its rapid and inexpensive deployment makes this type of network attractive in applications such as emergency rescue and battlefield communication.

IEEE 802.11 is an international WLAN standard that covers the specification for the medium access control (MAC) sublayer and the physical layer. An IEEE 802.11 WLAN can offer broadband data access at either 11 Mbps (IEEE 802.11b) or 54 Mbps (IEEE 802.11a). One commonly used function to coordinate channel access for STAs in the IEEE 802.11 standard is called the distributed coordination function (DCF), which is based on the CSMA/CA (carrier sense multiple access/collision avoidance) scheme.

In the DCF, any STA that is contending for a channel will execute the so-called binary exponential backoff procedure. A backoff counter is maintained at each contending STA, and its value is decreased for each channel idle time of one slot period. When the backoff counter reaches 0, an STA begins frame transmission. If a collision occurs, the STA chooses a larger counter value and executes the backoff procedure again. In the DCF, virtual carrier sensing is performed at the MAC sublayer by using the network allocation vector (NAV). Information on the duration of a frame exchange sequence between a transmission pair is included in the frame header and is announced to other STAs. Other contending STAs will wait for the completion of a current frame exchange by updating their NAVs.

In a wireless network environment, the hidden-terminal problem becomes problematic when two STAs attempt to send information to the same receiving STA. In the IEEE 802.11 standard, two STAs can avoid this problem by exchanging RTS (request to send) and CTS (clear to send) frames before transmitting the actual data frame. However, the DCF scheme still performs poorly in environments with high traffic loads, with frequent channel contentions and high collision rates limiting the throughput of an IEEE 802.11 WLAN.

The main factor underlying this weakness is all transmissions among STAs taking place on the same channel, with the resulting signal interference limiting network throughput. Actually, IEEE 802.11b and IEEE 802.11a systems contain maximum totals of 3 and 12 available nonoverlapping channels, respectively. Using multiple channels to remove the performance bottleneck of a single-channel network is promising, and some studies [1][2][3][4] have demonstrated the benefits of enabling multiple STAs in the same interference zone to transmit data simultaneously. However, the use of multiple channels results in some new challenges. When the STAs are located on different channels and are unaware of the channel activities of their neighbors, multichannel MAC protocols experience multichannel hidden-terminal, deafness, and broadcast problems.

The contribution of this paper is twofold. We first investigate the design of multichannel MAC protocols by classifying and comparing the advantages and disadvantages of the existing protocols in detail (Section 2). We then propose a hybrid protocol to enhance network throughput by tackling the weaknesses of existing protocols (Section 3). We evaluate the performance of the new protocol in Section 4, and draw conclusions in Section 5.

2. SURVEY OF RELATED WORK

The most important factor influencing the operation of a multichannel MAC protocol is the negotiation between the sender and receiver of a transmission pair for deciding the appropriate channel for data exchanges. These STAs need to decide when and where to perform negotiations, and how to agree on a common channel. We classify various multichannel MAC protocols into different categories according to the decision strategies used. Generally, the following two issues have to be considered in any design of this type of protocol:

1. Channel negotiation strategy: Decide how two arbitrary STAs whose current channel activities may be on different channels can be informed that they will next participate in the same communication.
2. Channel selection strategy: Decide on an appropriate channel from all available ones that benefits both the sender and the receiver.

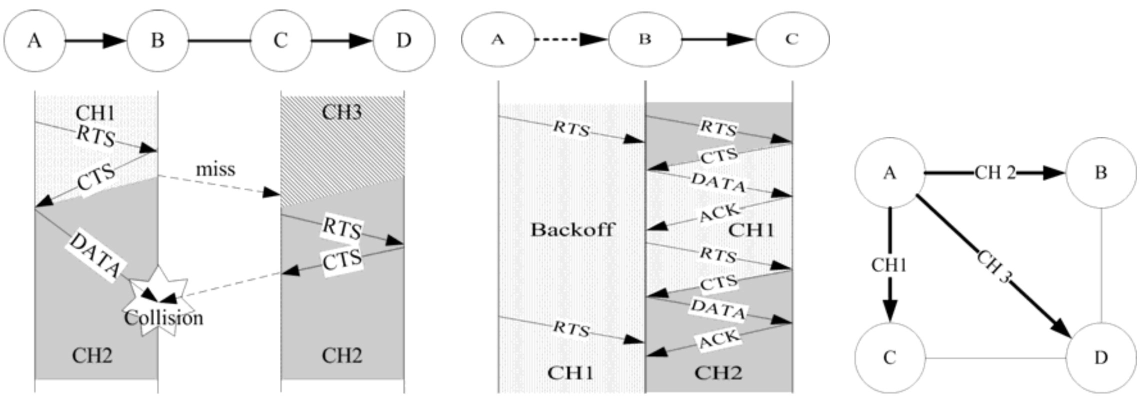
2.1. General Problems

Before introducing the various strategies, below we first discuss the problems that may be encountered in a multichannel design.

Multichannel hidden-terminal problem: An STA is called a multichannel hidden terminal if it interferes with one of its neighbors by attempting a transmission after switching to the same channel that this neighbor is currently using. An example to illustrate this problem is shown in Figure 1(a). Suppose that STA A has a packet for STA B. Both STAs exchange RTS/CTS frames on channel 1 and decide to use channel 2 for data exchange. Meanwhile, a neighboring STA C that operates on channel 3 does not know that channel 2 has been reserved and tries to negotiate with STA D to use the same channel 2 for data exchange. Hence, this problem can occur when one STA is not aware of a channel negotiation made by other STAs. One solution to this problem is to allow all STAs to continuously listen to all channel negotiations.

Deafness problem: The deafness problem occurs when an STA continuously attempts to contact another STA that is located on a different channel. One example is shown in Figure 1(b), where STA A tries to communicate with STA B, which is communicating with STA C on a different channel. STA A fails to receive any response from STA B and retries the transmission after a backoff period. Unfortunately, STA B has switched to a different channel before the end of the backoff period and STA A again receives no response. In an extreme case, this problem may cause a chain of STAs to be in waiting state. There are two types of solution: (1) an STA that is currently engaged in a transmission can use an additional radio transceiver to respond to transmission requests from other STAs, and (2) a sending STA will announce the completion of the current data exchange to its neighboring STAs. For example, STA B in the figure will continuously send a control packet to inform STA A of its being idle after it receives the ACK frame in channel 1. STA A can then stop the backoff and immediately send the RTS frame out. This solution does not require an additional transceiver.

Broadcast problem: If the neighboring STAs operate on different channels, it becomes impossible for an STA to send messages to them using broadcast. Figure 1(c) shows that this instead requires multiple unicast messages to be sent, which incurs a high cost. In multihop ad-hoc networks, the broadcast plays an important role in discovering routing paths (or neighbor discovery and topology discovery). One solution to this problem is to have a dedicated broadcast channel or to let all STAs switch periodically to a common channel.



(a) (b) (c)

Figure 1. Problems encountered in multichannel MAC protocols.

2.2. Channel Negotiation Strategy

The common knowledge plays a central role in any channel negotiation strategy. Each STA with common knowledge knows how to meet other STAs at what time and on what channel, using either see-all or visit-you approaches. In the see-all category, all STAs meet together at the same channel, and the used techniques include common control channel, common control period, and common hopping sequence. This category has been called a single rendezvous [3]. In the visit-you category, an STA voluntarily switches to the channel on which the intended receiving STA is currently active, and the used techniques include a private home channel and a private hopping sequence. This category has been called a multiple rendezvous [3]. Below we explain each of the techniques used in both of these categories.

Common control channel: In this scheme, one or more control channels are dedicated to the exchange of control packets for negotiations, and the remaining available channels (called data channels) are used for data exchanges. Any transmission pair can perform negotiations at any time on the dedicated control channel and then switch to a commonly selected data channel for data exchange. Examples of this category include DCA [5], RBCS [6], DPC [7], CSC [8], AMNP [9], Bi-MCMAC [10], MCDA [11], and MCMAC [12]. All protocols of this category except for MCMAC use only a single control channel.

Common control period: This scheme uses no dedicated control channel, with instead one data channel (called the common channel) temporarily acting as a control channel. All STAs will synchronously alternate between control and data periods. During a control period, all STAs have to switch to the common channel and perform negotiations with the destination STAs of pending data packets. After a successful negotiation, the two STAs of a transmission pair have to wait for the current control period to finish, and then switch to the commonly selected data channel for data exchange. Note that the common channel can also be selected as a data channel for data exchange. Examples of this category include MMAC [13] and MAP [14].

Common hopping sequence: The frequency hopping technique is used in this scheme, and all STAs hop through the available channels in the same sequence. When an STA has pending data to send, it will contact the intended receiving STA on the current channel. Both STAs stay on the same channel to exchange data, and then hop to the same channel as the remaining STAs after completing data exchange. One example of this category is HRMA [15].

Private home channel: In this scheme, each STA has a dedicated home channel to listen to, with these dedicated home channels being statically or dynamically assigned to STAs. A sending STA first learns the home channel of the intended receiving STA and then switches to this home channel when it has pending data to send. This information can be either broadcasted or piggy-backed with any outgoing packet from an STA. Examples of this category include HMCP [16], PCAM [17], and xRDT[18], where HMCP and xRDT transmit multiple unicast messages, and PCAM transmits broadcast messages on a dedicated broadcast channel to inform neighboring STAs of the channel settings.

Private hopping sequence: In this scheme, each STA has a dedicated hopping sequence, which can be generated by a common pseudo-random generator using a selected seed. Each STA has to inform neighboring STAs of its selected seed. Examples of this category include SSCH [19] and McMAC [20]. In McMAC, both the sending and receiving STAs stay on the same channel until the data exchange is completed, and then resume the individual hopping sequences. In SSCH, a sending STA changes its own hopping sequence to that of the intended receiving STA during the period of data exchange.

Below we compare the minimum number of transceivers necessary for each STA, the time synchronization requirement between all STAs, and the maximum number of channels that can be fully utilized together (called the channel bound in this paper) for each strategy.

Two transceivers are commonly employed when using a single common control channel: (1) one for keeping in touch with other STAs on the dedicated control channel, and (2) the other for dynamically switching to the selected data channel for data exchange. Some proposed protocols [8][9] have employed only one transceiver but with the limit of deferring every access for a certain time period when switching to a selected data channel. Moreover, the channel selected when using a single transceiver may be not suitable since an STA does not know the up-to-date channel status. In particular, MCMAC [12] needs m+1 transceivers, since m control channels are used. One transceiver is sufficient when using a common control period, with this transceiver having to alternate between control and data periods. This can be viewed as a slow frequency hopping system. One transceiver is also sufficient for other hopping-based approaches, such as those employing common/private hopping sequences. However, the hidden-terminal problem can occur and the deafness problem becomes worse when using a private hopping sequence and a single transceiver, since none of the STAs can obtain full information on the channel activities. The use of a private home channel suffers from the same problems for the same reason as the use of a private hopping sequence.

All hopping-based approaches are dependent on time synchronization, with tight synchronization needed in a fast frequency hopping system. Timing synchronization becomes an overhead in wireless ad-hoc networks. Moreover, the time and energy consumed on hopping might be significant.

We now compute the channel bound for each approach. We assume that the channel switch time is negligible. Let T_n and T_d respectively denote the average times to complete negotiations and data exchange for a transmission pair. T_n consists of the backoff time and the time to exchange control frames (e.g., RTS and CTS frames), and T_d is the time to exchange data and ACK frames. If $T_n < T_d$, using the common control channel would result in a maximum of $\lfloor T_d / T_n \rfloor$ transmission pairs being able to complete negotiations on the control channel during time T_d. Hence the channel bound for data channels is $\lfloor T_d / T_n \rfloor$, which means that adding more data channels than this value has no effect. Therefore, the total channel bound is $\lfloor T_d / T_n \rfloor + 1$ (including one control channel). Figure 2(a) shows an example where the total channel bound is 4, where the fourth transmission pair can use the same data channel as the first one.

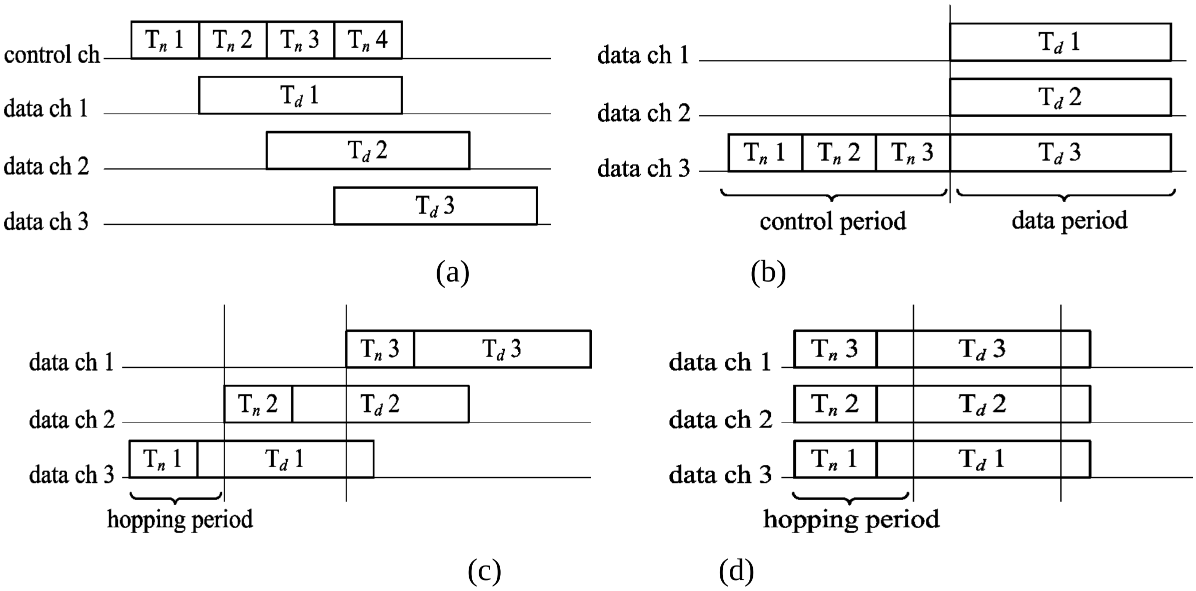


Figure 2. Utilization capacities for different approaches.

For the second approach, let the length of a control period be T_c. Since all negotiations have to be performed during T_c, the total number of transmission pairs that can complete negotiations is $\lfloor T_c/T_n \rfloor$. All these transmission pairs will simultaneously enter into the data period and will be distributed to at most $\lfloor T_c/T_n \rfloor$ data channels. Hence, the channel bound for data channels is $\lfloor T_c/T_n \rfloor$, this also represents the total channel bound since there is no control channel. Figure 2(b) shows an example where the total channel bound is 3.

For the common hopping sequence, let T_h be the time that an STA stays in each channel. During T_h, some of the STAs perform channel negotiations and data exchanges on the same channel. As in the example shown in Figure 2(c), if there are n transmission pairs that can complete negotiations, the following condition has to be satisfied:

$$nT_n + (n-1)T_d \le T_h \le (n+1)T_n + nT_d \quad \text{or}$$

$$n(T_n + T_d) \le T_h + T_d \le (n+1)(T_n + T_d)$$

Hence we get $n = \lfloor (T_h + T_d)/(T_n + T_d) \rfloor$. These n transmission pairs will occupy the channel for time $n(T_n + T_d)$, during which other STAs can hop to at most $\lceil n(T_n + T_d)/T_h \rceil$ channels (including the initial one). Hence, the total channel bound is also equal to this value.

For visit-you-based approaches, channel negotiations can occur on different channels simultaneously. Hence, all available channels can be utilized together, as shown in the example in Figure 2(d). The total channel bound therefore equals the total number of available channels in the system (denoted by N).

Table 1. Features of different channel negotiation strategies.

Category	Approach	Transceiver#	Synchronization	Channel Bound
See-all based	Common control channel	≥2	No	$\lfloor T_d/T_n \rfloor + 1$
	Common control period	≥1	Yes (weak)	$\lfloor T_c/T_n \rfloor$
	Common hopping sequence	≥1	Yes (tight)	$\lceil n(T_c + T_d)/T_h \rceil$, $n = \lfloor (T_h + T_d)/(T_c + T_d) \rfloor$
Visit-you based	Private home channel	≥1	No	N
	Private hopping sequence	≥1	Yes (tight)	N

In summary, if there are infinite available channels, the channel bound is limited for all approaches of the see-all category, but is unlimited for those approaches of the visit-you category. We conclude the above observations in Table 1. In Table 2, we summarize the advantages and disadvantages of each approach.

2.3. Channel Selection Strategy

In the see-all category, all protocols based on the common control channel or the common control period encounter the decision problem of how to agree on a common channel during negotiations. Frequency-hopping-based protocols avoid this problem. The allocation of private home channels to individual STAs can also be viewed as a decision problem. However, many protocols use a fixed and random allocation for simplicity.

We classify the various channel selection strategies into global and local schedules, where each STA knows the channel activities of all other STAs in the global schedule, but only the channel activities of neighboring STAs in the local schedule. Examples of the global schedule include MAP [14] and MAXM [21]. The MAP protocol is based on the approach of using a common control period during which each STA collects all negotiation data. This means that every STA has a global view about how many transmission pairs will take place and how long each transmission pair will spend on the channel. Based on this information, every STA uses the same shortest-job-first scheduling algorithm to arrange these transmission pairs onto appropriate channels. After the control period, each transmission pair will switch to the correct channel based on the schedule. Constructing a unique global schedule requires all STAs to collect the same data, which also requires that all STAs are within a single hop distance (although the optimal schedule can be found, this method cannot be applied to multihop network environments). The MAXM protocol uses the maximal matching algorithm to determine the channel selection. Although this protocol can be implemented in a distributed manner, this would require the exchange of many messages between neighboring STAs.

Almost all of the other protocols use local scheduling because of its low maintenance overhead. We consider the design of a local schedule based on the selection criterion (the information referred to in the decision) and decision maker (whether the sender or receiver makes the decision).

Table 2. Comparisons between different approaches.

Approaches	Advantages	Disadvantages
Common control channel	● The dedicated control channel can serve as a broadcast channel. ● The channel negotiations and transmissions of broadcast packets can be performed at any time, and hence the queuing delay of pending packets is small.	● The utilization of the dedicated control channel on which only control packets are transmitted may be low. Moreover, this channel represents the bottleneck of the entire system. ● The channel bound is limited and is relevant to the bandwidth of the dedicated control channel.
Common control period	● The common control period is a suitable time for broadcasting any message. ● The common channel can be used to transfer data packets during data periods, which increases channel utilization.	● Time synchronization is needed among STAs. ● The channel negotiations and transmissions of broadcast packets cannot be performed at any time, and hence the queuing delay of pending packets is large. ● The channel bound is limited and is relevant to the length of a control period.
Common hopping sequence	● The broadcast problem is absent. ● The channel negotiations and transmissions of broadcast packets can be performed at any time. ● The hopping nature can reduce the interference of channel errors.	● Time synchronization is needed among STAs. ● Frequent channel switching might result in a high power consumption. ● The channel bound is limited and is relevant to the hopping frequency.
Private home channel	● The channel negotiations can be performed simultaneously on different channels, which can increase the system throughput and make channel bound unlimited.	● The broadcast problem is present. ● The multichannel hidden-terminal problem can occur, with the resulting collisions reducing the system throughput. ● The deafness problem becomes more serious if only one transceiver is used. ● It takes extra cost for each STA to learn the channel settings of its neighboring STAs.
Private hopping sequence	● This has the same advantages as the private home channel. ● The hopping nature can reduce the interference of channel errors.	● This suffers from the same problems as the private home channel. ● Time synchronization is needed among STAs. ● Frequent channel switching might result in a high power consumption.

There are three criteria commonly used in channel selection: idle state (e.g., DCA [5], RBCS [6], DPC [7], AMNP [9], MCMAC [12], and MAP [14]), traffic load (e.g., MMAC [13] and HMCP [16]), and random assignment (e.g., MCDA [11]). When using the idle state, the channel that first becomes idle is selected; when using the traffic load, the channel that has the lightest traffic load is selected; and when using random assignment, the channel is selected at random. Using the first two criteria will incur an information collection overhead (i.e., all the negotiation data), while using the last criterion may result in poor performance.

We further explore the approach of using the idle state. The idle state of a channel can be detected using either physical or virtual channel sensing. In physical sensing, an STA determines whether a channel is currently idle by switching the transceiver to that channel and detecting whether a physical signal is present. MCSMA [22] deploys N transceivers, each of which listens to a distinct channel (given that there are N available channels). This protocol avoids the use of channel switching, but clearly has a high hardware cost. Moreover, the use of physical sensing results in the hidden-terminal problem, since a neighboring STA of a receiving STA cannot sense the signal transmitted from the sending STA. Some methods avoid this problem by using a busy-tone technique, in which a jamming signal is transmitted continuously on a control channel by both the sender and receiver during the period of data exchange on a data channel. Any possibly hidden STA can listen to the control channel to learn the state of the corresponding data channel. This approach has two disadvantages: (1) at least two transceivers are needed at each STA, and (2) every data channel has to be allocated a control channel. An example protocol is xRDT [18]. Considerable channel resources and device energy are wasted, but the strength of this type of protocol is that any neighboring STA can correctly and immediately learn the idle state of a data channel.

Virtual channel sensing is based on estimating how long a data channel is used by a transmission pair and is similar with the setting of the NAV. DCA [5] uses this method to maintain the release time of each data channel. The drawback of this type of protocol is that collisions may occur if the release times maintained in each STA are not consistent due to node mobility or packet errors.

No collisions would occur for a transmission pair directly commencing data exchange on the selected channel if perfect idle states are maintained. However, several transmission pairs might select the same channel when using traffic load or random assignment. A collision avoidance mechanism such as the backoff procedure should be performed (possibly with the exchange of RTS/CTS frames) before the actual data transmission. Moreover, if these transmission pairs switch to the same channel at different times, STAs with later incoming transmissions might become hidden terminals for the current ongoing transmission pair. One solution is to delay any channel access for an STA with a newly incoming transmission for a period equal to the maximum time required to complete a data exchange sequence.

The final decision on channel selection during negotiations may be made by a sender, a receiver, or a hybrid of a sender and receiver. Irrespective of which scheme is employed, some new frame structures are defined below for carrying negotiation data.

Sender-based scheme: In this scheme, the sender always decides on the data channel to use for data exchange with a receiver according to its own selection criterion. The receiver may either conditionally or unconditionally accept the decision. If the receiver accepts unconditionally, this type of protocol needs another channel contention on the selected data channel to prevent collision at the receiver side. If the receiver accepts conditionally, it can reject the decision if its own selection criterion is violated. This may involve several rounds of negotiations between the sender and receiver, each of which involves a two-way handshake using the RTS–CTS frame sequence. DPC [7] and MCDA [11] are examples of this scheme.

Receiver-based scheme: In this scheme, a sender tells its corresponding receiver the state (idle state or traffic load) of each data channel it knows about. Then the receiver compares this received information with its own information, and then selects a data channel that is suitable for both sides. If neighbors on the sender side need to be informed of the decision result, these negotiations will involve a three-way handshake using the RTS–CTS–RES frame sequence. The RES (reserve) frame is a newly defined frame for conveying the final agreement of negotiations. The transmission of channel states is an overhead in this scheme, examples of which are DCA [5], MCMAC [12], and MMAC [13].

Hybrid-based scheme: In this scheme, a sender tells its corresponding receiver both the state of each data channel and a single recommended data channel. If the receiver accepts the recommendation, this scheme works as the sender-based scheme; otherwise it works as the receiver-based one. An example of this scheme is AMNP [9].

2.4. Discussion

We summarize the principles of multichannel design in Figure 3. The multichannel MAC protocols proposed in recent years and the principles they use are listed in Table 3.

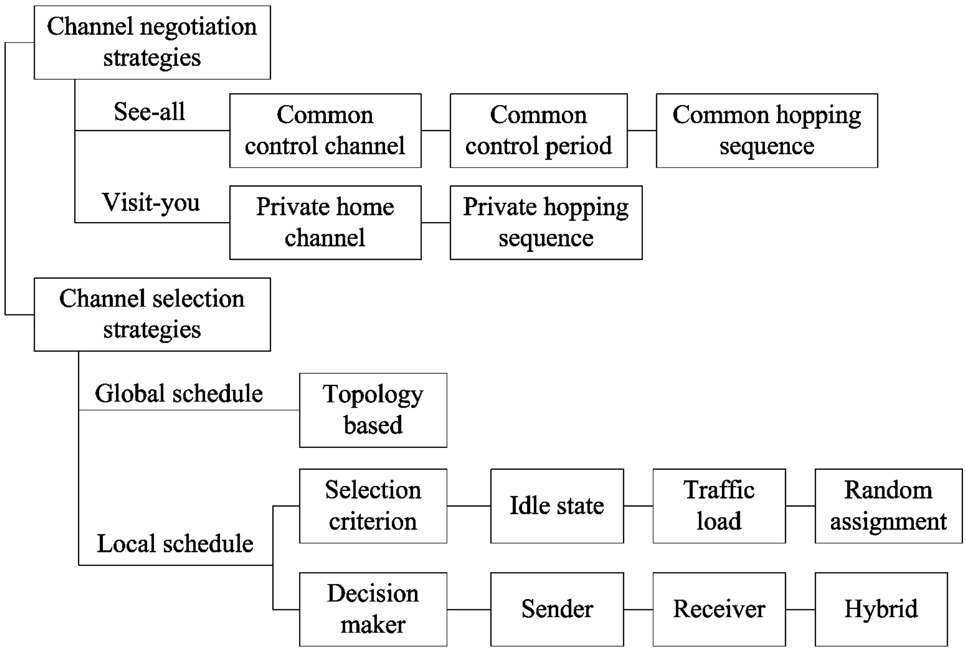


Figure 3. Design principles for multichannel MAC procotols.

Table 3 indicates that most of the protocols belong to the see-all category. Although the visit-you-based protocols exhibit a promisingly high channel bound, Section 2.2 indicates that they suffer from many problems. If the hardware cost is not a major concern and each STA can be equipped with more than one transceiver, the common control channel is a good candidate for channel negotiations and broadcast data transmissions. The features of broadcast are important in highly dynamic multihop ad-hoc networks, since broadcast facilitates the routing process. The local schedule is suitable for a large-scale network. Using traffic load as a selection criterion has the advantage of easy maintenance (in MMAC, the traffic load of a channel is determined by simply counting the number of transmission pairs that have selected this channel) relative to using the release time. One drawback of using traffic load is that an additional channel contention is needed when an STA switches to a different channel. However, this channel contention is necessary even using other selection criteria for robustness in mobile and error-prone wireless environments. The receiver-based channel decision is of greater concern, since the correctness of data transmission largely depends on the receiver side. In a single-hop network, the sender-based decision is sufficient because both the sender and receiver experience similar channel conditions. Finally, the flexibility of the hybrid scheme means that it can be used in any condition.

Table 3. Summary of existing protocols.

Protocol	Channel negotiation	Transceiver#	Selection criterion	Decision maker
DCA [5]	Common control channel	2	Idle state (virtual sensing)	Receiver
RBCS [6]	Common control channel	2	Idle state (virtual sensing)	Receiver
DPC [7]	Common control channel	2	Idle state (virtual sensing)	Sender
CSC [8]	Common control channel	1	Idle state (virtual sensing)	Receiver
AMNP [9]	Common control channel	1	Idle state (virtual sensing)	Hybrid
Bi-MCMAC [10]	Common control channel	1	Idle state (virtual sensing)	Receiver
MCDA [11]	Common control channel	2	Random	Sender
MCMAC [12]	Common control channel	m+1	Idle state (virtual sensing)	Receiver
MMAC [13]	Common control period	1	Traffic load	Receiver
MAP [14]	Common control period	1	Idle state (virtual sensing)	Global schedule
HRMA [15]	Common hopping sequence	1	Hopping sequence	N/A
HMCP [16]	Private home channel	2	Receiver's channel assignment	N/A
PCAM [17]	Private home channel	3	Receiver's channel assignment	N/A
xRDT [18]	Private home channel	2	Receiver's channel assignment	N/A
SSCH [19]	Private hopping sequence	1	Receiver's channel assignment	N/A
McMAC [20]	Private hopping sequence	1	Receiver's channel assignment	N/A
MCSMA [22]	N/A	N	Idle state (physical sensing)	Sender

Finally, we illustrate the current trend on multichannel MAC protocol design. Some research work such as [23] considers the problem under wireless mesh networks with tree topology and emphasizes a cross-layer design between MAC and routing protocols. A major routing path should experience a less number of channel switches for time and energy efficiency. If the channel switch time becomes significant, the paper [24] found an interesting phenomenon: All routing paths that are intersected with each other prefer using the same channel on whole paths. A more complex version of the problem is to consider the channel allocation with multiradio and multichannel environments

3. PROPOSED PROTOCOL

We now demonstrate the power of combining the common control channel and the common control period – by mixing the DCA [5] and MMAC [13] protocols. The initial ideas are introduced in our previous paper [25]. In our proposed protocol, each STA requires two transceivers: (1) the control transceiver, which permanently operates on the dedicated control channel, and (2) the data transceiver, which can switch to different data channels.

Our protocol includes power management. In the ad-hoc mode of an IEEE 802.11 WLAN, each STA can go into sleep mode to save power and wake up periodically at each ATIM window. During an ATIM window, a sender instructs the intended receiver to stay awake. In MMAC, the ATIM window is explained as the common control period. Indeed, the tasks of waking up and negotiating with other STAs can be combined. Other non-control-period protocols do not take advantage of these ATIM windows.

In contrast to MMAC, we use two transceivers to perform negotiations during an ATIM window on two channels. This doubles the channel bound, which means that we can achieve the same channel bound as MMAC whilst using half an ATIM window and leaving more time for data transmission. Moreover, we can also arrange one transceiver to perform negotiations and the other one to perform data broadcast or multicast during the ATIM window. During each non-ATIM window period, our proposed protocol behaves like those protocols using the common control channel but with the enhancements described below.

First, we enable the control channel to be a data channel by allowing the control transceiver to also transmit data packets. In the extreme case, an STA can be concurrently involved in separate data transmissions on the control and data channels, which would facilitate bidirectional data transmission.

Second, we consider the data retransmission issue, which has been seldom discussed in the literature. In DCA, a data retransmission has to be renegotiated on the control channel. This would increase the overhead on the control channel, particularly when channel errors are numerous. We propose performing data retransmission on the same selected channel without any extra negotiation.

We use traffic load as a selection criterion and the hybrid sender and receiver as a decision maker. Moreover, we take the multirate feature of wireless systems into account when counting channel traffic loads. The distances between STAs connected in an ad-hoc network may differ, with closer STAs being able to exchange data at a higher data rate. Therefore, each link in the topology of such a network may have a different data capacity. We measure the traffic load of a channel by summing the total channel occupation times taken by those current transmission pairs on that channel. The data capacity of each link can be determined by a cross-layer design with a quality-of-service-based protocol that decides routes according to link quality. Each STA can select an appropriate data rate for each of its links to neighboring STAs. During the negotiation phase, the channel decision maker estimates the channel occupation time using the following formula, and piggybacks this information with the control packets:

Occupation time = (total size of pending packets to send)/(link data rate of the selected channel).

Neighboring STAs can determine the traffic load by monitoring these control packets.

We now explain how negotiation is achieved in our protocol. A sender normally sends an RTS frame to the intended receiver to initiate the negotiation. This RTS frame also states the size of pending packets to send. The receiver responds to the sender with a CTS frame that states the selected channel number with the lowest traffic load and the estimated channel occupation time. Then the sender sends out an RES frame to confirm the agreement and also inform the neighboring STAs of the selected channel and its occupation time. At that moment the sender and receiver can switch to the selected channel. The sender then executes the backoff procedure to contend for channel access. If the sender switches to a different channel from the currently used one, there is a time delay equal to the maximum time to complete a data exchange sequence (i.e., one data frame and one ACK frame) before the sender proceeds further. A sender terminates a data exchange sequence with the receiver when the final ACK frame is successfully received or the maximum number of retries has been reached. A receiver assumes that the data exchange sequence has finished when the next ATIM window begins.

A pure sender (i.e., which is not a receiver in any other transmission) can agree with different receivers on using different data channels. When one data exchange sequence has finished, the sender switches to the designated channel to serve the next data exchange. However, a pure receiver or a hybrid sender and receiver (being a sender in one data exchange sequence and a receiver in another) can only agree with other STAs on using the same data channel during the negotiations in the same ATIM window. That is, the data transceiver in this case cannot be switched dynamically. A simple example for understanding this limitation is when two senders make contact with the same receiver. The receiver cannot stay on a different channel at a different time, and we do not know which sender will begin transmission first. A nonpure sender will force the intended receiver to accept the channel selected by the sender (if any), or the receiver will reject the requirement if there is another selected channel that has been decided upon by the receiver. A nonpure sender will also reject the decision made by a receiver if it has agreed with another STA on using a different channel.

The detailed operations of the proposed protocol are described below:

Step 1 (if power management is enabled): Each STA should wake up when entering an ATIM window and tune the data transceiver into a default data channel (commonly with all STAs). An STA can perform negotiations with other STAs using both transceivers simultaneously. If the control channel is selected for data exchange, the control transceiver is used; whereas if a data channel is selected, the data transceiver is used. Both transceivers can operate simultaneously if necessary.

Step 2: Any data transmission begins after an ATIM window has finished. A sender can send more than one data frame to the receiver. If any transmission error occurs, an STA performs a data retransmission on the same channel. The retransmission for the same data packet can be repeated until the maximum retry limit is reached. The receiver will remain on the same channel until the beginning of the next ATIM window.

Step 3: An STA can initiate negotiations on the control channel during non-ATIM window periods for newly pending data packets. If the data transceiver of either the sender or receiver is in a busy state due to an incomplete data exchange, the control transceiver is used to serve data exchanges on the control channel.

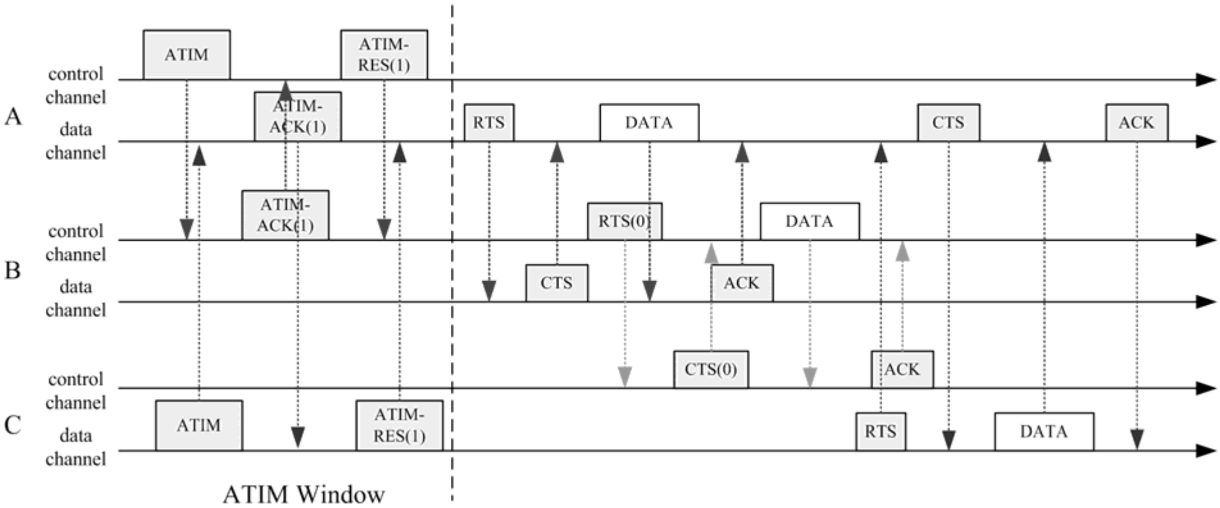


Figure 4. A running example. The parenthesized numbers indicate whether a control channel (0) or a data channel (1) is selected.

A scenario example is shown in Figure 4, in which only two channels (one control channel and one data channel) are available. During the ATIM window, the control frames used are ATIM, ATIM-ACK, and ATIM-RES. In this example, STA A informs STA B of pending data packets on the control channel and STA B decides to use the data channel. Also, STA C does the same thing with STA A but on the data channel. In this case, STA A is a hybrid sender and receiver and can switch to a single channel only. These data transmissions take place on the data channel after the ATIM window by following the RTS–CTS–DATA–ACK frame sequence. In the non-ATIM window period, STA B has pending data packets for STA C and hence the negotiation is performed on the control channel. Since STA B has an incomplete data exchange sequence, STA B forces STA C to use the control channel for data exchange.

4. PERFORMANCE EVALUATION

We have written simulation programs using the CSIM [26] tool. Here we compare the performance of our protocol (abbreviated as “Hybrid”) with DCA and MMAC in a single interference zone. That is, we consider a fully connected topology where all STAs are able to transmit to each other. Also, we disable power management, and count the number of transmission pairs as the traffic load in our scheme to allow a fair comparison with DCA and MMAC. The arrival of data packets at a sender is modeled as in a typical Web browsing application by using the Weibull distribution. We assume that each data packet has a size of 1024 bytes for simplicity. The network parameters follow the IEEE 802.11 family standards, as listed in Table 4. The wireless channel condition is modeled as a two-state Markov process with good and bad states. The duration of these two states is exponentially distributed with means 30 ms and 10 ms. The bit error rates (BERs) of these two states, BER_{good} and BER_{bad}, are assumed to be 10^{-10} and 10^{-4}, respectively. In MMAC, the control and data periods are 20 ms and 60 ms, respectively.

Table 4. Simulation settings.

Parameter	Value
Transmission pairs	10–80 (default, 80)
No. of channels	2–10 (default, 3)
Channel rate	2 Mbps
Simulation time	60 s
Slot time	20 μs
SIFS	10 μs
DIFS	50 μs
CW, max	1023
CW, min	15
Retry limit	6

In the simulation, we use the following metrics to measure the performance: Utilization is the average throughput of each channel.

$$Throughput = \frac{Packet\_Length \times No\_Successful\_Packets}{Total\_SimTime}$$

$$Average\_Delay = \frac{Total\_Packet\_Delay}{No\_Successful\_Packets}$$

$$Drop\_Rate\ (\%) = \frac{No\_Dropped\_Packets}{No\_Total\_Generated\_Packets}$$

$$Utilization = \frac{Packet\_Length \times No\_Successful\_Packets}{Total\_SimTime \times No\_Channels}$$

Figure 5 shows throughput comparisons when varying the number of transmission pairs. The network load increases with the number of transmission pairs. The protocols exhibit similar performance when the network load is low, but as this nears saturation, Hybrid performs significantly better than both DCA and MMAC. This is because the available bandwidth of the control channel is fully utilized by the proposed protocol. Hybrid uses the counter-based channel selection, which can simplify the channel decision process and can balance the channel load on the network. In contrast, DCA has to compute the release time of each data channel and agree on a free channel before data transmission. Moreover, only one data packet can be transmitted after negotiations. This results in complex channel negotiations and a wasteful reservation process when data transmission fails on the selected data channel. MMAC has the penalty of the control period during which data packets cannot be transmitted. Even though we have selected the best size for the control period in the simulation, its throughput is still lower than that of Hybrid.

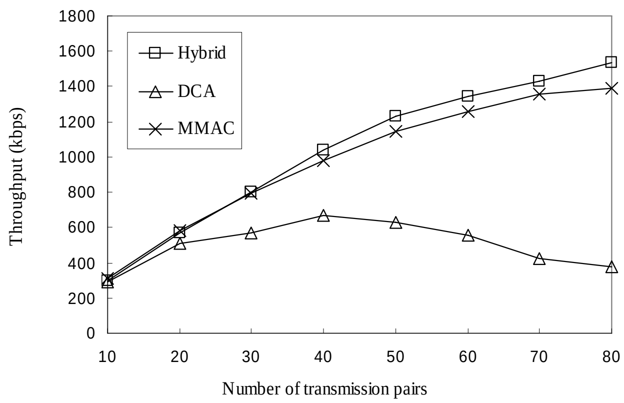


Figure 5. Throughput vs. number of transmission pairs.

Figure 6 plots the average packet delay as a function of the network load, which shows that this is similar for DCA and Hybrid. MMAC has a higher delay, since STAs have to wait for the current control period to finish before beginning to transmit data packets. Moreover, not every transmission pair can complete channel negotiations during the control period, and these pairs have to wait for the next control period, which further postpones packet transmissions.

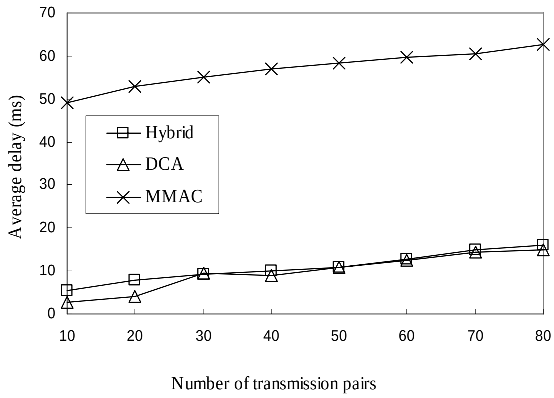


Figure 6. Average delay vs. number of transmission pairs.

Figure 7 compares the drop rates of the different protocols, which is lowest for Hybrid. DCA suffers from a high drop rate because of the large number of contentions (per-packet negotiations and renegotiations) on the control channel.

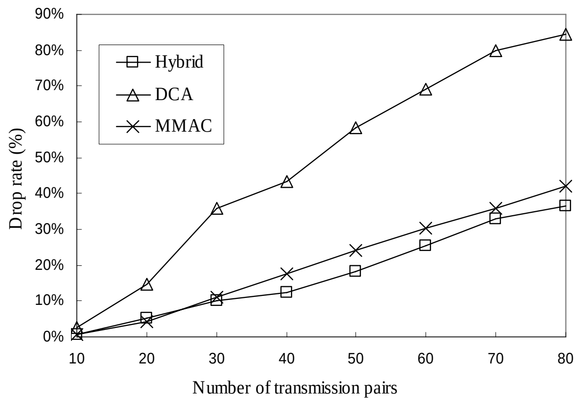


Figure 7. Drop rate vs. number of transmission pairs.

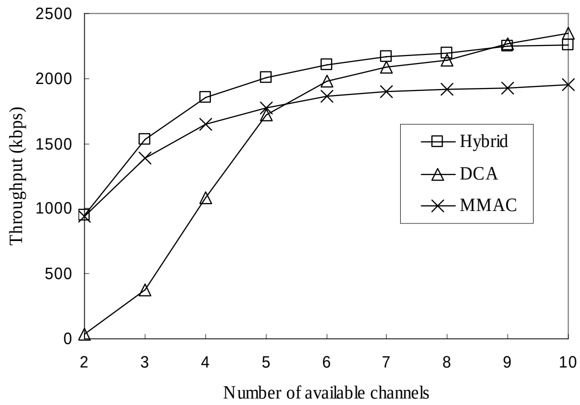


Figure 8. Throughput vs. number of channels.

Figure 8 presents throughput comparisons when varying the number of available channels, for 80 transmission pairs. Hybrid performs better than DCA when few channels are available, which is due to Hybrid having an additional channel (the control channel) to use as a data channel. However, DCA shows a large improvement as the channel number increases, and may even slightly outperform Hybrid for a large number of channels. This is because the use of the control channel for data transmission reduces the opportunities for channel negotiations in Hybrid – we found that Hybrid performs similarly to DCA when many channels are available if we disable the use of the control channel as a data channel. MMAC performs worst among all when channel number is greater than six in our experiment. This reveals the fact that common-channel-based approaches can accommodate more transmission pairs than common-period-based ones.

Figure 9 compares utilizations when varying the number of available channels, again for 80 transmission pairs. The channel utilization is higher for Hybrid than for the other protocols. Hybrid and DCA can fully utilize three and five channels, respectively, under the same traffic load. However, MMAC reaches the top utilization at two channels. Adding more channels provides no benefit, since the maximum number of transmission pairs that can successfully complete negotiations is bounded.

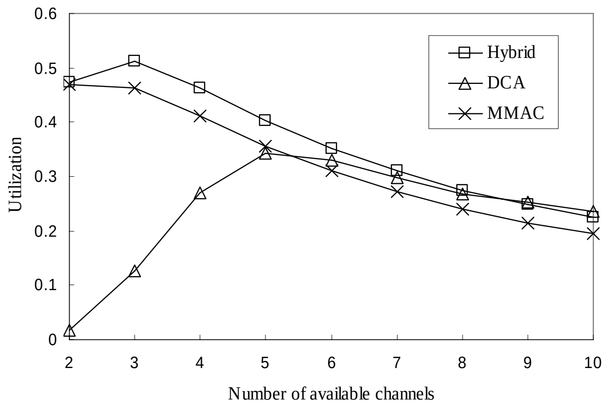


Figure 9. Utilization vs. number of channels.

5. CONCLUSIONS

In this paper, the design of multichannel MAC protocols in wireless ad-hoc networks is discussed. The multichannel feature can increase the network performance, but introduce multichannel hidden-terminal, deafness, and broadcast problems as well. We investigate the existing work and extract the common techniques that have been exploited in previous studies. We identify two categories (see-all and visit-you) and five approaches (common control channel, common control period, common hopping sequence, private home channel, and private hopping sequence) for use in channel negotiations. We show that the channel bound is limited for all approaches of the see-all category, but is unlimited for those approaches of the visit-you category. However, the visit-you-based protocols cannot fully combat with the problems in multichannel design. Most of the existing protocols employ local scheduling to determine the channel usage. We found that using traffic load is more robust and flexible as a criterion on channel selection than using idle state (or release time), particularly in mobile and error-prone environments.

We then presented a multichannel MAC protocol that is simple yet efficient. Our proposed protocol combines the advantages of using the common control channel and the common control period, and also introduces some new enhancements on data retransmissions, bi-directional and multirate communications. We demonstrated that this protocol can achieve higher throughput and channel utilization than two representative protocols (DCA and MMAC). Two issues need to be considered in the future: (1) the protocol design in an environment where STAs are equipped with different numbers (or types) of transceivers, and (2) the design of the wireless mesh networks, which have become popular in recent years.

REFERENCES

[1] P. Kyasanur and N. Vaidya, “Capacity of Multi-Channel Wireless Networks: Impact of Number of Channels and Interfaces,” International Conference on Mobile Computing and Networking (MobiCom), pp. 43-57, Aug. 2005.

[2] A. Baiocchi, A. Todini, and A. Valletta, ”Why a Miltichannel Protocol can Boost IEEE 802.11 Performance,” ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM), pp. 143-148, Oct. 2004.

[3] J. Mo, H. S. W. So, and J. Walrand, “Comparison of Multi-Channel MAC Protocols,” ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM), pp. 209-218, Oct. 2005.

[4] P. Kyasanur, J. So, C. Chereddi, and N. H. Vaidya, “Multichannel Mesh Networks: Challenges and Protocols,” IEEE Wireless Communications, vol. 13, no.2, pp. 30-36, Apr. 2006.

[5] S. L. Wu, C. Y. Lin, Y. C. Tseng, and J. P. Sheu, “A New Multi-Channel MAC Protocol with On-Demand Channel Assignment for Multi-Hop Mobile Ad Hoc Networks,” International Symposium on Parallel Architectures, Algorithms and Networks (I-SPAN), Dec. 2000.

[6] N. Jain, S. R. Das, and A. Nasipuri, “A Multichannel CSMA MAC Protocol with Receiver-Based Channel Selection for Multihop Wireless Networks,” Computer Communications and Networks, pp. 15-17, Oct. 2001.

[7] W. C. Hung, K. Law, and A. Leon-Garcia, “A Dynamic Multi-Channel MAC for Ad Hoc LAN,” Biennial Symposium on Communications, Apr. 2002.

[8] N. Choi, Y. Seok, and Y. Choi, “Multi-Channel MAC Protocol for Mobile Ad Hoc Networks,” IEEE Vehicular Technology Conference (VTC), pp. 1379-1382, Oct. 2003.

[9] J. H. Chen and Y. D. Chen, “AMNP: Ad Hoc Multichannel Negotiation Protocol for Multihop Mobile Wireless Networks,” IEEE International Conference on Communication (ICC), pp. 3607-3612, Jun. 2004.

[10] T. Kuang and C. Williamson, “A Bidirectional Multi-Channel MAC Protocol for Improving

TCP Performance on Multihop Wireless Ad Hoc Networks," ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM), pp. 301-310, Oct. 2004.

[11] C. Y. Chang, H. C. Sun, and C. C. Hsieh, "MCDA: An Efficient Multi-channel MAC Protocol for 802.11 Wireless LAN with Directional Antenna", International Conference on Advanced Information Networking and Applications (AINA), pp.64-67, Mar. 2005.

[12] H. Koubaa, "Fairness-Enhanced Multiple Control Channels MAC for Ad Hoc Networks," IEEE Vehicular Technology Conference (VTC), pp. 1504-1508, Jun. 2005.

[13] J. So and N. Vaidya, "Multi-Channel MAC for Ad Hoc Networks: Handling Multi-Channel Hidden Terminals Using A Single Transceiver," International Symposium Mobile Ad Hoc Networking and Computing (MobiHoc), pp. 222-233, May 2004.

[14] J. Chen, S. T. Sheu, and C. A. Yang, "A New Multi-Channel Access Protocol for IEEE 802.11 Ad Hoc Wireless LANs," Personal, Indoor and Mobile Radio Communications, IEEE, pp.7-10, Sept. 2003.

[15] Z. Tang and J. J. Garcia-Luna-Aceves, "Hop-Reservation Multiple Access (HRMA) for Ad Hoc Networks," IEEE International Conference on Computer Communications and Networks (IC3N), pp. 388-395, 1998.

[16] P. Kyasanur and N. Vaidya, "Routing and Interface Assignment in Multi-Channel Multi-Interface Wireless Networks," IEEE Wireless Communications and Networking Conference (WCNC), pp. 2051-2056, Mar. 2005.

[17] J. S. Pathmasuntharam, A. Das, and A. K. Gupta, "Primary Channel Assignment Based MAC (PCAM) - A Multi-Channel MAC Protocol for Multi-hop Wireless Networks," IEEE Wireless Communications and Networking Conference (WCNC), pp. 21-25, Mar. 2004.

[18] R. Maheshwari, H. Gupta, and S. R. Das, "Multichannel MAC Protocols for Wireless Networks," IEEE Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON), Sept. 2006.

[19] P. Bahi, R. Chandra, and J. Dunagan, "SSCH: Slotted Seeded Channel Hopping for Capacity Improvement in IEEE 802.11 Wireless Networks," International Conference on Mobile Computing and Networking (MobiCom), pp.216-230, Oct. 2004.

[20] H. W. So and J. Walrand, "McMAC: A Multi-Channel MAC Proposal for Ad-Hoc Wireless Networks", Technical Report, Berkeley, Apr. 2005.

[21] S. H. Hsu, C. C. Hsu, S. S. Lin, and F. C. Lin, “A Multi-Channel MAC Protocol Using Maximal Matching for Ad Hoc Networks,” International Conference on Distributed Computing Systems Workshops (ICDCSW), pp. 505-510, 2004.

[22] A. Nasipuri, J. Zhuang, and S. R. Das, “A Multi-Channel CSMA MAC Protocol for Multi-Hop Wireless Networks,” IEEE Wireless Communications and Networking Conference (WCNC), pp.1402-1406, Sept. 1999.

[23] A. Raniwala, K. Gopalan, and T. C. Chiueh, “Centralized Channel Assignment and Routing Algorithms for Multi-Channel Wireless Mesh Networks,” Mobile Computing and Communication Review, vol. 8, no. 2, pp. 50-65, Apr. 2004.

[24] R. Vedantham, S. Kakumanu, S. Lakshmanan, and R. Sivakumar, “Component Based Channel Assignment in Single Radio, Multi-channel Ad Hoc Networks,“ MobilCom, pp. 378-389, Sept. 2006.

[25] S. C. Lo and C. W. Tseng, "A Novel Multi-Channel MAC Protocol for Wireless Ad Hoc Networks," IEEE Vehicular Technology Conference (VTC-Spring), pp. 46-50, Apr. 2007.

[26] The Mesquite Software Inc., “The User’s Guide of CSIM Simulation Engine,” http://www.mesquite.com/.