

A Combined Proactive Routing and Multi-Channel MAC Protocol for Wireless Ad Hoc Networks

Michelle X. Gong Scott F. Midkiff Shiwen Mao
The Bradley Department of Electrical and Computer Engineering
Virginia Polytechnic Institute and State University, Blacksburg, VA 24061 USA
{mgong, midkiff, smao}@vt.edu

Abstract— To improve the capacity of wireless ad hoc networks by exploiting multiple available channels, we propose a combined proactive routing and multi-channel medium access control (MAC) protocol. The multi-channel MAC protocol is compatible with IEEE 802.11 MAC and imposes the minimum system requirement among existing multi-channel MAC protocols. Because a proactive routing protocol allows each node to have complete topology information of the network, channel assignment can be closely coupled with a proactive routing protocol. The proposed channel assignment protocol is shown to require fewer channels and exhibit significantly lower communication, computation, and storage complexity than existing channel assignment schemes. We prove the correctness of the proposed channel assignment protocol. In addition, through a performance study, we show that the proposed protocol, by effectively increasing capacity, substantially increases throughput and decreases delay compared to the IEEE 802.11 MAC protocol.

I. INTRODUCTION

Despite recent advances in wireless local area network (WLAN) technologies, today's WLANs still cannot offer the same data rates as their wired counterparts. The throughput problem is further aggravated in multi-hop wireless environments due to *intra-flow interference* introduced by adjacent nodes on the same path and *inter-flow interference* generated by nodes from neighboring paths. For instance, it has been pointed out by Li, *et al.* [12] that the maximum capacity that the IEEE 802.11 MAC can achieve for a chained network is just one seventh of the available bandwidth.

All current IEEE 802.11 physical (PHY) standards divide the available frequency into several orthogonal channels, which can be used simultaneously within a neighborhood. Therefore, increasing capacity by exploiting multiple channels becomes particularly appealing. In fact, such bandwidth aggregation has been widely used in infrastructure-based WLANs, where high-end access points have multiple interfaces that operate on different channels simultaneously [13]. In infrastructure-based WLANs, non-overlapping channels are distributed among different access points at the network planning stage. However, IEEE 802.11 WLANs that operate in ad hoc mode rarely use multiple channels simultaneously partly because the IEEE 802.11 MAC is not designed to operate with multiple channels. Therefore, an ad hoc network that is based on IEEE 802.11a technology utilizes only one out of 12 available orthogonal channels, wasting more than 90 percent of the potentially available spectrum.

Consequently, there has been substantial interest in multi-channel MAC schemes that can achieve higher throughput by exploiting multiple available channels [8], [13], [20]. Some of the early works, e.g. [16] and [9], assume that every node has its own unique channel. Therefore, no channel assignment or selection is needed. However, in reality, the number of channels is limited and has to be carefully assigned to each node to avoid contention and collisions and to enable optimal spatial re-use of available channels. Many distributed channel assignment problems are proven to be \mathcal{NP} -complete and, thus, computationally intractable [2], [7]. There exist only a few heuristic solutions, which have good performance under certain environments, for instance, in a static wireless network. However, they suffer from inefficiency when employed in the mobile ad hoc environment [4], [7].

In this paper, we propose a combined proactive routing and multi-channel MAC protocol that utilizes an efficient channel assignment algorithm. This efficient channel assignment algorithm is based on three design principles [6]. First, to reduce the complexity of the channel assignment algorithm, channel assignment and routing should be jointly considered. This "cross-layer" design approach is motivated by the fact that both the channel assignment algorithm and the ad hoc routing algorithm must be invoked when there is a change in the network topology. Exploring this design principle can greatly reduce the complexity of channel assignment algorithms. Secondly, channels should be assigned only to "active" nodes. This "on-demand" channel assignment principle is motivated by the fact that only active nodes need valid channels. Fewer channels may be required in the network if this "on-demand" assignment principle is implemented. Finally, both primary and secondary collisions and interference should be taken into consideration. To mitigate interference as well as to resolve collisions, distinct channels should be assigned in a way that collisions and interference can be avoided as much as possible.

We propose to combine channel assignment with a proactive ad hoc routing protocol called Optimized Link State Routing (OLSR) [3]. While the proposed Channel Assignment OLSR (CA-OLSR) assigns distinct channels to active nodes, a Multi-Channel MAC (MC-MAC) protocol is utilized to manage multi-channel medium access. In general, wireless ad hoc routing protocols can be classified into two categories: *reactive* and *proactive*. In a reactive routing protocol, nodes do not have complete network topology information, channel assignment

can be based only on partial topology information [6]. On the other hand, a proactive routing protocol allows each node to have the complete up-to-date topology information of the network based on periodic exchanges of control messages. Therefore, proactive routing protocols can be closely coupled with channel assignment, without causing significant modifications to the protocols themselves.

As a result of the cross-layer approach, CA-OLSR exhibits significantly lower communication, computation, and storage complexity than existing channel assignment protocols. Following the on-demand channel assignment principle, CA-OLSR is invoked only when there is a route change, which further reduces control overhead. In addition, unlike some channel assignment schemes that ignore a node's one-hop neighbors [4], [7], CA-OLSR assigns distinct channels to active nodes within a k -hop neighborhood to avoid both primary and secondary collisions as well as to mitigate interference.

The rest of the paper is organized as follows. We formulate the channel assignment problem in Section II. In Section III, we describe the proposed CA-OLSR protocol and give an overview of the MC-MAC protocol. A performance study is presented in Section IV. Section V gives an overview of related work on distributed channel assignment and multi-channel MAC protocols. Finally, Section VI concludes this paper.

II. PROBLEM FORMULATION

A wireless ad hoc network can be modelled as an undirected graph $G = \{V, E\}$, where V is the set of nodes and E is the set of edges that represent links. We assume that nodes use omnidirectional antennas and radio links are bidirectional. A link is assumed to exist between two nodes if and only if the two nodes are within each other's radio range.

Two or more wireless nodes may generate *primary collisions* if they are one hop away, while *secondary collisions* can be generated by nodes that are two hops away. Such collisions can be eliminated if active nodes within a two-hop range of each other transmit on different orthogonal channels. Furthermore, we define the interference range to be the k -hop neighborhood of a node. Interference can be significantly reduced if nodes within the k -hop neighborhood are assigned different orthogonal channels. To avoid collisions as well as to mitigate interference, we propose to assign distinct channels on-demand to "active" nodes in a k -hop neighborhood. Parameter k is a user-defined neighborhood size that is constrained by the number of available channels. A larger k translates into a higher demand on available channels, for the same signal-to-interference-noise-ratio (SINR). In general, the neighborhood size k should be chosen as large as possible, constrained by the number of available channels, to mitigate interference and to enable effective channel re-use in the network.

Before we formulate the distributed channel assignment problem, we define the k -hop neighbors of a node v to be the set $N_k(v) = \{w \in V | h(v, w) \leq k\}$, where $h(v, w)$ is the hop distance from v to w , i.e. the minimum length of any path from v to w . For instance, $N_1(v)$ is the set of one-hop neighbors of node v . \mathcal{C} denotes the set of all available

channels in the network. We further define $V_t \subset V$ to be the set of active transmitters and $V_r \subset V$ to be the set of active receivers. Note that a node cannot be both a transmitter and a receiver at the same time, i.e. membership in V_t and V_r is mutually exclusive. Let $v_{r,i} \in V_r$ be a particular receiver and $v_{t,j} \in V_t$ be a particular transmitter. If $v_{t,j}$ is transmitting on the same channel on which $v_{r,i}$ is receiving, but $v_{t,j}$ is not the intended transmitter to $v_{r,i}$, then transmitter $v_{t,j}$ will cause interference at receiver $v_{r,i}$. The power associated with the interference, $P(v_{t,j}, v_{r,i})$, is the received power at node $v_{r,i}$, which is a function of the distance between nodes $v_{t,j}$ and $v_{r,i}$, the transmitting power, and the channel condition, e.g. path loss. Table I summarizes the notation used.

TABLE I
NOTATION

Symbol	Comments
\mathcal{C}	The set of available channels
V	The set of nodes in the network
E	The set of edges that represent radio links
$v \in V$	A node in the network
$V_t \subset V$	The set of active transmitters
$V_r \subset V$	The set of receivers
$v_{t,j} \in V_t$	A particular transmitter
$v_{r,j} \in V_r$	A particular receiver
$v_{t,T(i)}$	The desired transmitter
β	The desired SINR threshold for successful packet reception
N_0	Additive white Gaussian noise (AWGN)

If k is set to an appropriate value and $|\mathcal{C}|$ is large enough, both primary and secondary collisions can be largely avoided and harmful interference can be mitigated. If the number of available channels is sufficient, distributed channel assignment algorithms should assign distinct channels to any nodes within a k -hop neighborhood. However, in many cases, the number of available channels may be less than the nodes in a k -hop neighborhood. Therefore, the primary design goal of distributed channel assignment is to minimize the maximum number of nodes sharing the same channel with any designated node $v_{t,j} \in V_t$ among this node's k -hop neighbors.

Minimize:

$$\max n_k(v_{t,j}), \quad \forall v_{t,j} \in V_t \quad (1)$$

subject to:

$$\frac{P(v_{t,T(i)}, v_{r,i})}{\sum_{v_{t,j} \in V_t \setminus v_{t,T(i)}} [P(v_{t,j}, v_{r,i}) \cdot S(v_{t,T(i)}, v_{t,j})]} + N_0 \geq \beta, \quad (2)$$

where N_0 is AWGN noise and β is the minimum SINR required for a successful packet reception. The term $S(v_{t,T(i)}, v_{t,j})$ indicates the relationship between the channels used by the desired transmitter $v_{t,T(i)}$ and transmitter $v_{t,j}$. Function $S(v_{t,T(i)}, v_{t,j}) = 0$ if $v_{t,T(i)}$ and $v_{t,j}$ are using different (strictly orthogonal) channels and $S(v_{t,T(i)}, v_{t,j}) = 1$ if $v_{t,T(i)}$ and $v_{t,j}$ are using the same channel. In CDMA

systems, different channels with non-orthogonal codes may be used, in which case $0 < S(v_{t,T(i)}, v_{t,j}) < 1$. In Equation (2), note that node $v_{t,T(i)}$ is the intended transmitter for the receiver $v_{r,i}$ and is excluded from the summation of interference sources.

The constraint means that the accumulated interference generated by active transmitters sharing the same or interfering data channels as the designated transmitter should be less than a certain threshold to ensure that the receiver can decode the data packet successfully. To meet the primary design goal and to satisfy the design constraint, a channel assignment protocol should distribute available channels with any pre-defined value of k in a way that the maximum number of transmitters that share the same data channel is minimized. Meanwhile, the same set of channels should be re-used in a way that the accumulated interference generated on any particular data channel is below a certain threshold.

III. DESCRIPTION OF THE COMBINED CA-OLSR AND MC-MAC PROTOCOL

In this section, we will first describe the proposed CA-OLSR protocol in detail and then provide a brief overview of the MC-MAC protocol.

A. Channel Assignment and Optimized Link State Routing protocol (CA-OLSR)

OLSR is a proactive routing protocol. As the name suggests, OLSR is essentially a link state routing protocol with an optimized flooding method that can reduce communication overhead [3]. OLSR minimizes the overhead of control packet flooding by using only selected nodes, called multipoint relays (MPRs), to retransmit control messages. Because not all nodes are involved in the flooding of control traffic, OLSR can reduce the number of retransmissions required to diffuse a message through the network. Secondly, OLSR requires only partial link state to be flooded to provide shortest path routes. Therefore, the size of control packets can be reduced. Each node selects its MPR set among its one hop neighbors in such a way that the set covers all the two-hop neighbors. For instance, the black nodes in Figure 1 are MPRs selected by the node in the middle. Once a node is selected as a multipoint relay, it not only retransmits routing control messages, but also serves as an intermediate node on routing paths. A route is essentially a sequence of hops through the multipoint relays from source to destination. Two types of control messages are defined in OLSR: HELLO message and topology control (TC). To select MPRs and calculate its routes to all known destinations through the MPRs, a node must first detect its one-hop and two-hop neighbor nodes by periodically broadcasting HELLO messages, which have a Time-To-Live (TTL) value of 1. HELLO messages permit each node to learn the topology of its neighbors up to two hops away. Based on this information, each node in the network independently selects its own set of MPRs that covers all the two-hop neighbors. MPRs of a given node are then declared in the subsequent HELLO messages so that the information can reach MPRs themselves. A neighbor

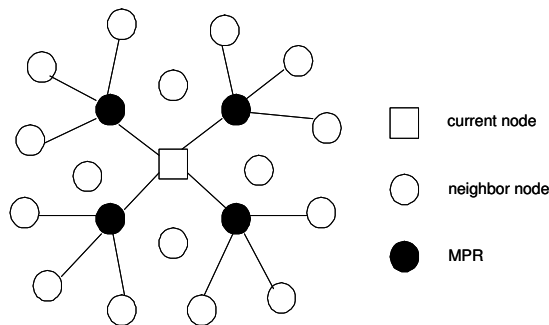


Fig. 1. Illustration of multi-point relays

table and an MPR selector table are constructed by each node using information obtained from the HELLO message.

A TC message is sent periodically by each node to declare its MPR selector set that consists of the list of neighbors who have selected the sender node as an MPR. TC messages are forwarded by nodes in the MPR set to the entire network. Each node maintains a topology table to record the topology information obtained from the TC messages. A TC message is larger than a HELLO message and is sent out less frequently than the HELLO message. A routing table is constructed at each node based on the information contained in the neighbor table and the topology table. The calculation of the routing table can follow any standard link state algorithm. An example procedure is given in RFC 3626 to explain how to calculate the routing table [3].

The basic ideas of CA-OLSR are to utilize routing control messages to exchange channel information and to assign distinct channels to active transmitters within a k -hop neighborhood. Because MPRs are the nodes that act as intermediate routes for other nodes, it is intuitive to give MPRs higher priority when the number of channels is insufficient to ensure no interference by all nodes. However, at any time, not all of MPRs are active. Moreover, nodes that do not act as MPRs may become active source nodes. To take these situations into consideration, we propose to identify active nodes within a certain time period and then assign a higher priority to active nodes when there is a channel conflict. Because HELLO messages are sent more frequently and can carry channel information up to two hops away, we choose HELLO messages to carry channel information. The neighborhood size in CA-OLSR is defined to be $k = 2$, based on the assumption that interference range is twice the radio range.

At initialization, each node in the network randomly chooses a channel from a set of all available channels, denoted by \mathcal{C} , and then sends out a HELLO message that carries its own channel index. After a node v has sensed its one-hop neighbors, the HELLO message will contain a list of its one-hop neighbors $N_1(v)$. A node should then append the channel information of its active neighbors in future HELLO messages. Nodes detect active neighbors by listening on the control channel. Neighbors that exchange RTS and CTS control messages in one hello interval are considered active neighbors, which are

```

Procedure sendHello ()
HELLO.myChannelIndex = myChannel;
for  $i=1$  to  $k$ 
    HELLO.MPRChannelIndex[  $i$  ] = MPRChannel[  $i$  ];
endfor
Broadcast HELLO
endprocedure

```

Fig. 2. Procedure sendHello()

then indicated in the HELLO messages sent in the next hello interval. If a node has selected its MPR set $M \in N_1(v)$, it should also indicate its MPRs in the HELLO message. Upon receiving HELLO messages from neighboring nodes, a node first checks to see whether there is a neighbor set contained in the message. If so, the node builds an available channel list, \mathcal{A} , by marking channels that are taken by active neighbors as unavailable.

If there is a channel conflict between the current node and an active neighbor, the current node should choose another channel from the available channel set \mathcal{A} . This is to ensure that active nodes have higher priority to obtain distinct channels than other nodes when the number of available channels is fewer than the number of nodes in the two-hop neighborhood. If there is a channel conflict between two active nodes, the node with the lower node ID retains its channel while the other node should mark the channel-in-conflict as unavailable and randomly pick a new channel from its updated available channel set \mathcal{A} . The same procedure applies when two inactive nodes have a channel conflict. In general, active nodes have higher priority over in-active nodes. Within the set of active or inactive nodes, the nodes with lower node IDs have higher priority.

The sendHello() and recvHello(HELLO) procedures are summarized in Figure 2 and Figure 3. In the recvHello(HELLO) procedure, the *getActiveNeighborInfo(HELLO)* function retrieves channel information from HELLO messages received from neighboring nodes, while the *randomChannel(A)* function returns a channel index that is randomly chosen from the available channel set \mathcal{A} . The channel assignment procedure in CA-OLSR is invoked “on-demand”, only when a channel conflict occurs after a topology change. Further, CA-OLSR seeks to assign distinct channels to active nodes in the two-hop neighborhood to minimize both primary and secondary collisions.

B. Overview of the MC-MAC Protocol

Most current multi-channel MAC protocols need two functionalities: channel assignment and medium access control [8], [13], [20]. The primary goal of multi-channel MAC design is to achieve high channel throughput and to reduce delay by maximizing spatial re-use of multiple channels and minimizing contention and collisions on each individual channel. To achieve this goal, the channel assignment function tries to utilize as many channels as possible within a neighborhood and tries to keep the channel conflicts to a minimum. Meanwhile, the medium access control function makes sure that even when

```

Procedure recvHello (HELLO)
neighborChannels = getActiveNeighborInfo(HELLO);
Update the available channel set  $\mathcal{A}$ ;
channelConflict = FALSE;
if Detect a channel conflict
    if (activeNode == TRUE)
        if (activeMe == FALSE)
            channelConflict = TRUE;
        else if (NodeID < myID)
            channelConflict = TRUE;
        endif
    endif
else
    if (activeMe == FALSE & NodeID < myID)
        channelConflict = TRUE;
    endif
endif
if channelConflict == TRUE
    myChannel = randomChannel (A);
endif
endprocedure

```

Fig. 3. Procedure recvHello(HELLO)

two nodes in the same neighborhood select the same channel, contention and collisions on the channel can be effectively controlled and resolved. Because the channel assignment is performed by CA-OLSR, our MC-MAC protocol only needs to manage medium access control on multiple data channels. MC-MAC is essentially compatible with the single-channel IEEE 802.11 MAC due to the following reasons.

- Like the 802.11 MAC, MC-MAC is based on carrier-sense multiple access with collision-avoidance (CSMA/CA).
- MC-MAC requires only one half-duplex transceiver per host and does not require synchronization among nodes. This is exactly the same system requirement as for an 802.11 node. To the best of our knowledge, MC-MAC protocol has the minimum system requirement among all multi-channel MAC protocols proposed to date.
- Like the 802.11 MAC, MC-MAC only needs to perform medium access control, whereas most other multi-channel MAC protocols must also perform channel assignment.

We assume that there is one dedicated control channel and up to N data channels in the network. Each data channel is equivalent and has the same bandwidth. We further assume that each host in the network is equipped with only a single half-duplex transceiver, which means that nodes cannot receive on different channels simultaneously and that a transmitting node is not able to receive other transmissions.

MC-MAC is a transmitter-based protocol. Nodes are assigned channels by the CA-OLSR routing protocol. When a node is ready to transmit, it first informs the destination node of its assigned data channel. As shown in Figure 4, when the sender node A intends to transmit, it first uses the control channel to broadcast a request-to-send (RTS) message, carrying its own data channel index, c_i . Upon receiving the RTS message, the destination node B uses the control channel

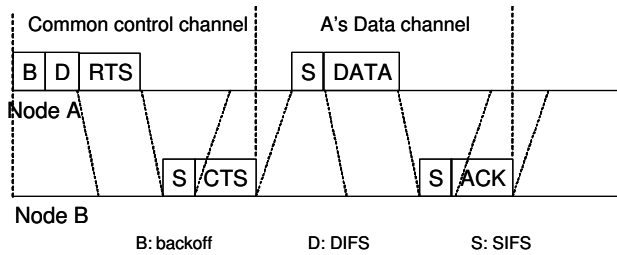


Fig. 4. Four-way handshake procedure of MC-MAC

to send back a clear-to-send (CTS) message carrying c_i and switches its receiving channel to c_i . After node A receives the CTS message, it switches to the designated data channel and starts data transmission. The sender listens to the data channel until an ACK is received or a timeout occurs. Similarly, after sending the ACK message on the data channel, the receiving node also switches back to the control channel. In general, when a host is not engaged in transmitting a data packet or receiving a data packet, it should always listen on the control channel.

IV. CORRECTNESS OF CA-OLSR

Under the assumption that the number of available channels is sufficient and the channel information is updated more often than the topology changes, we show that CA-OLSR can assign distinct channels to any active node within a two-hop range.

For the purpose of analysis only, CA-OLSR is assumed to operate under the following conditions.

- 1) The number of available channels is more than the number of active nodes within a two-hop range at any time.
- 2) All HELLO messages transmitted over a radio link are received correctly in the same hello interval.
- 3) Topology and channel changes can be conveyed to all nodes within a two-hop neighborhood before the next topology change. This could be achieved by choosing an appropriate HELLO interval according to the current mobility level.

Here, we assume that HELLO messages are sent frequently enough that a change in topology can be detected by nodes in the two-hop range within one or two hello intervals. This is also to assume that the frequency of topology changes is less than the topology update frequency.

Theorem 1: Within a finite time after a topology change, CA-OLSR can assign distinct channels to any active nodes within a two-hop neighborhood.

To prove the above statement, we need to show that the following statements are true.

- 1) CA-OLSR is dead-lock free and live-lock free.
- 2) Within a finite amount of time after a topology change, all nodes must have consistent and up-to-date topology and channel information about nodes within its two-hop neighborhood.
- 3) If the information in the node is consistent and up to date, then CA-OLSR can assign channels correctly, such

that no two active nodes in the two-hop neighborhood share the same data channel.

Lemma 1: CA-OLSR is dead-lock free and live-lock free. In addition, the channel updating procedure will take at most $l - 1$ steps, where $l \geq 2$ is the number of active nodes that are involved in a channel conflict.

Proof: The proof of the dead-lock free property of CA-OLSR is straight forward. In CA-OLSR, each node in the network is assigned only one channel. However, for deadlocks to occur, a node must hold at least one resource (i.e. a channel) before it requesting another resource. Since CA-OLSR assigns only one channel to each node, there is no deadlock.

Next, we prove that CA-OLSR is also live-lock free. In CA-OLSR, a node updates its data channel only if the following three conditions are satisfied at the same time.

- 1) There is a channel conflict between a node and at least one of its one-hop or two-hop neighboring nodes.
- 2) At least one neighboring node that has the same channel as the current node is active.
- 3) The node ID of the active neighbor is less than that of the current node.

Three scenarios may occur where at least one node in the network needs to switch its data channel.

Case I: Assuming that there are exactly two active nodes that choose the same data channel, one node must have a node ID that is less than that of the other node because node IDs are unique within the network. Then, the node with lower node ID keeps its data channel while the node with higher node ID updates its own data channel to avoid the channel conflict. In this two-node case, no live-lock will occur.

Case II: Assuming that there are $l > 2$ active nodes that pick the same data channel, there must be a node that has the lowest node ID among the l active nodes. The node with the lowest node ID keeps its data channel, while each of the rest $l - 1$ nodes pick a new data channel for itself. Note that because the original data channel has been marked as unavailable at each of the $l - 1$ nodes, all of them pick a data channel that is different from the original one. In the worst-case situation where all $l - 1$ nodes pick the same data channel again, the node that has the lowest node ID among the $l - 1$ nodes can retain its data channel while the other $l - 2$ nodes randomly pick a new data channel from the available channel set \mathcal{A} . The available channel set \mathcal{A} at each node is updated at each step. The channel update procedure continues until all l nodes have distinct data channels. In the worst case, this takes $l - 1$ steps. Because at every step the node that can retain its original channel is determined by its ID, which is unique in the network, no live-lock can occur either.

Case III: In the case that there are two or more different channel conflicts that involve a total of l active nodes, the same upper bound still holds. First, because each node is assigned only one channel at any time, different channel conflicts always involve different nodes. Thus, different channel conflicts can be resolved independently in parallel. The worst-case scenario occurs when at the second step, all remaining nodes

pick the same data channel. Then, this situation degenerates into Case II.

As can be seen from the above three cases, no deadlock and live-lock can occur in CA-OLSR. In addition, the channel update procedure takes at most $l - 1$ steps, where l is the number of active nodes that have a channel conflict. ■

Corollary 1.1: The value of the Hello interval should be $3(l-1)$ times smaller than the topology change interval, where l is the maximum number of neighbors that can pick the same channel at the same time.

Proof: After updating its own data channel, a node v should send the updated topology information and channel information to its one-hop neighbors $N_1(v)$ in either the current hello interval, h_0 , or the next hello interval, h_1 . Note that if the change of channel happens after a node sends out a HELLO message, the change will be conveyed to its one-hop neighbors in the next hello interval. To get the upper bound, we assume that the updated HELLO message is always sent out in the second hello interval h_1 . Upon receiving the updated HELLO message, one-hop neighbors $N_1(v)$ of the current node convey the updated channel information to the two-hop neighbors $N_2(v)$ by sending their HELLO messages in the next hello interval. Therefore, a channel update may take up to three hello intervals to reach all the nodes within a two-hop range.

The proof of Lemma 1 shows that it takes at most $l - 1$ steps to resolve channel conflicts that involves l active nodes. To complete the channel update procedure, the updated channel information may need to propagate to all nodes in the two-hop neighborhood at each step. Since each step of the channel update procedure takes up to three hello intervals, the whole procedure takes up to $3(l - 1)$ hello intervals. Because the channel update procedure has to be completed before the next topology change, the hello interval should be $3(l - 1)$ times shorter than the topology change interval. ■

Lemma 2: Within a finite amount of time after a topology change, all nodes have consistent and up-to-date topology and channel information about nodes within its two-hop neighborhood.

Proof: Consistent and up-to-date topology and channel information means that a node knows the most recent topology change and all the recent channel changes among its one-hop and two-hop neighbors.

We know from Lemma 1 that CA-OLSR does not deadlock or live-lock. Specifically, the channel update procedure terminates after at most $l - 1$ steps, where l is the maximum number of active nodes in the two-hop neighborhood that pick the same data channel. Because the number of nodes in the two-hop neighborhood is finite, l must be finite. Further, because we assume that the topology change is slow enough that topology update and channel update information can be conveyed to all nodes within the two-hop neighborhood before the next topology change, the next topology change will not interfere with the channel update procedure invoked by the current topology change. Therefore, within a finite time, the channel updating procedure terminates and all nodes eventually stop

updating their available channel set \mathcal{A} and their own data channels.

All HELLO messages transmitted over a radio link are assumed to be received correctly in the same hello interval. Thus, at most three hello intervals after the channel selection algorithm terminates, HELLO messages can propagate to any node's two-hop neighbors. Therefore, within a finite amount of time after a topology change, all nodes must have consistent and up-to-date topology and channel information about nodes within their two-hop neighborhood. ■

Lemma 3: After the channel update procedure terminates, no two active nodes in the two-hop neighborhood share the same data channel.

Proof: The channel update procedure terminates when nodes do not make any new channel update. Since the topology and channel information in each node are consistent and up-to-date, as stated in Lemma 2, the channel update procedure terminates based on the correct topology and channel information. Because the channel update procedure terminates only when there is no channel conflict within any two-hop neighborhood, no two active nodes in the two-hop neighborhood share the same data channel. ■

Based on the proven lemmas, we conclude that within a finite time after a topology change, CA-OLSR can assign distinct channels to any active nodes within a two-hop neighborhood. Thus, Theorem 1 is proved.

V. PERFORMANCE EVALUATION

In this section, we demonstrate the capacity improvement of CA-OLSR combined with MC-MAC over the standard OLSR with the original IEEE 802.11 MAC and MC-MAC with randomly assigned channels, via ns-2 simulations [19].

We assume that 64 wireless nodes are placed randomly in

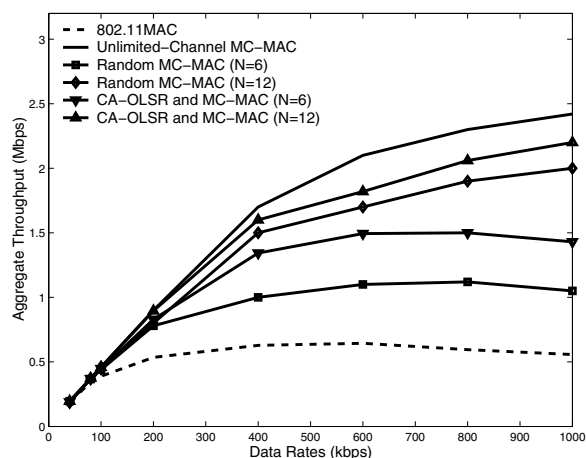


Fig. 5. Aggregate throughput in a dense network

an area. Each node is equipped with only one half-duplex transceiver. All nodes in the network share the same common control channel. There are 6 or 12 different data channels available. In all simulations, the radio range of a node is set to 250 m and the interference range is set to 550 m, which is

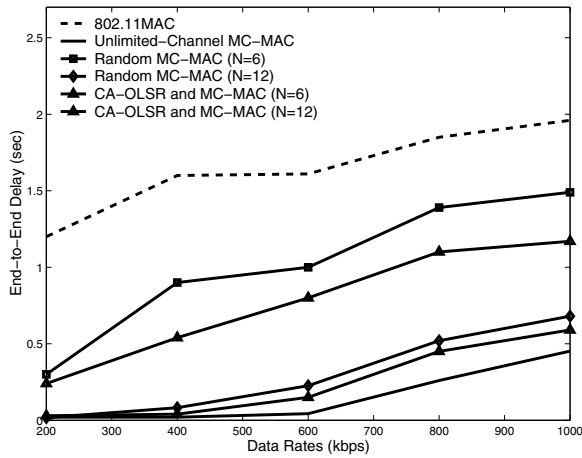


Fig. 6. Average end-to-end delay in a dense network

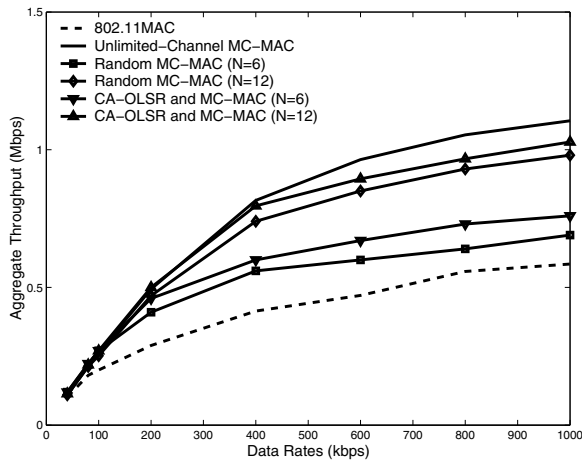


Fig. 7. Aggregate throughput in a sparse network

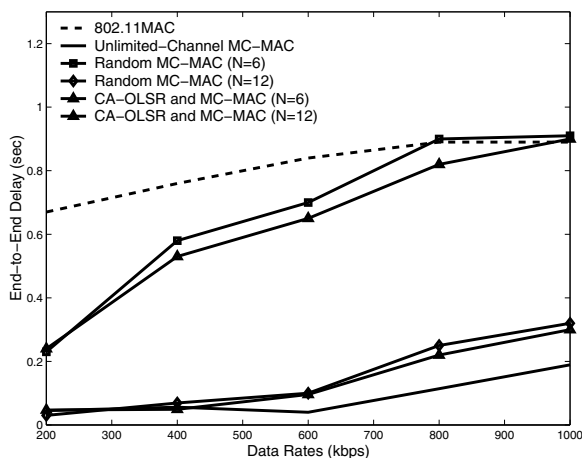


Fig. 8. Average end-to-end delay in a sparse network

approximately twice the radio range. The two-ray ground path loss model is used. The physical layer data rate for all data channels and the control channel is set to 2 Mbps [19]. For routing, normal OLSR is used with the IEEE 802.11 MAC, while CA-OLSR is used with MC-MAC. Six UDP flows are generated in the network. Each UDP flow has an offered load ranging from 40 kbps to 1000 kbps. The channel switch delay is $80 \mu\text{s}$ [1]. Simulation period is set to be 7600 seconds, where the warm-up period is 3600 seconds and the effective simulation time is 4000 seconds.

The first network is a 800m by 800m dense network and the second network is an 1600m by 1600m sparse network. Mobile nodes move randomly according to the random waypoint mobility model, where the maximum speed is 5 m/s and the minimum speed is 4 m/s. The maximum pause time is 5 s. Although we have also studied CA-OLSR at higher speeds, the results are not significantly different. The network size has a much greater impact on the performance than node speed. We compare CA-OLSR with a scheme that utilizes unlimited number of data channels and a random-OLSR algorithm. In the unlimited data channel scheme, each node has its own unique data channel in the network and a common control channel is shared by all nodes in the network. In the random-OLSR algorithm, a node determines its own channel index based on its MAC address. Therefore, no channel assignment is needed in the random-OLSR scheme.

Figure 5 shows that CA-OLSR always performs better than the random-OLSR scheme in a dense network, given an equal number of channels. The performance gap between the two increases as the number of available channels decreases. Note that CA-OLSR combined with MC-MAC can have up to 4 times higher throughput than the IEEE 802.11 MAC. Figure 6 shows that the end-to-end delay increases for all schemes when the data rate increases. However, multi-channel schemes that utilize 12 data channels have much lower delay than the original 802.11 MAC scheme.

Figure 7 and Figure 8 show the throughput and delay for all schemes in a sparse network. The performance gaps

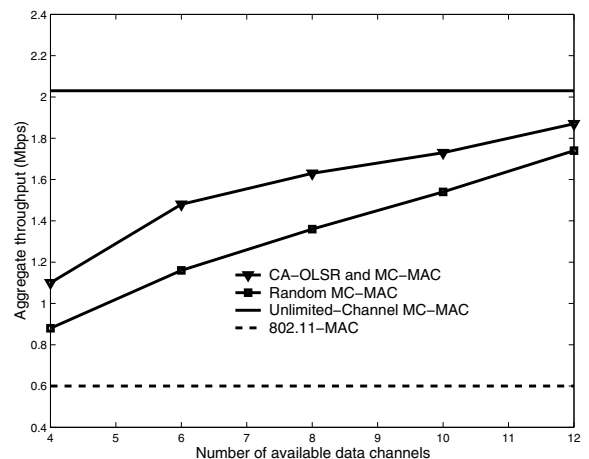


Fig. 9. Impact of the number of channels on the aggregate throughput

between different schemes are not as large as those in the dense network, although CA-OLSR still has an advantage over the random-OLSR scheme. In a sparse network, the interference and collisions generated by neighboring nodes is lower compared to those in a dense network because in a sparse network, each node has fewer neighbors and the distance between neighbors is greater. Therefore, the performance gain of multi-channel MAC schemes is not as significant in a sparse network. Figure 9 shows the performance improvement with an increase in the number of available data channels. The performance gap between CA-OLSR and random-OLSR decreases when the number of available channels increases.

Based on the simulation results, we make the following observations.

1) As expected, regardless of the channel assignment schemes, utilizing multiple channels always gives better performance than the IEEE 802.11 MAC scheme. Utilizing multiple channels increases the capacity available in an ad hoc network.

2) In all simulated scenarios, the performance gap between the unlimited channel scheme and CA-OLSR with 12 data channels is not significant. Thus, we can conclude that due to the negligible interference generated by nodes that are far away, a large number of data channels is not necessary to gain benefits.

3) The performance gap between CA-OLSR and the random scheme decreases when the number of available channels increases. CA-OLSR assigns channel intelligently. We showed in Section IV that CA-OLSR can assign distinct channels to any active node in a two-hop neighborhood when the number of available channels is sufficient. Therefore, CA-OLSR can effectively avoid primary collisions and secondary collisions. The random scheme merely tries to utilize all available channels without intelligent channel assignment. However, when there is a large number of available channels, collisions may be infrequent even if channels are assigned randomly.

4) The performance advantage of CA-OLSR combined with MC-MAC over the IEEE 802.11 MAC scheme is not in proportion to the number of channels utilized. For instance, for MC-MAC with 12 available data channels, the throughput gain can be up to 4 times, compared to the throughput achieved by the IEEE 802.11 MAC. The question is whether there is a way to get 12 times performance gain when there are 12 available data channels. The answer is yes, if there are sufficient number of transceivers at each node and traffic flows can utilize the channels.

Most current mobile devices are equipped with only one half-duplex transceiver. We want to study the performance gain that an off-the-shelf mobile device can obtain using multiple available channels. Therefore, we assume the same hardware requirement for both MC-MAC and the IEEE 802.11 MAC. For MC-MAC, each node is assumed to be equipped with only one half-duplex transceiver, which is time-shared between transmission and receiving, and between the common control channel and the data channel. With an increase in the number of data channels, the time period allocated to the

common control channel also increases because the number of collisions on the common control channel increases as more nodes try to transmit in parallel. Therefore, the time period a transceiver can spend on transmitting data packets is reduced, which degrades the performance advantage of CA-OLSR. In fact, Kyasanur and Vaidya [10] shows that when the number of transceivers is less than the number of available channels, network capacity degrades.

VI. RELATED WORK

The prior work relevant to CA-OLSR and MC-MAC can be divided into two categories: distributed channel assignment and multi-channel MAC protocols.

A. Related Work on Distributed Channel Assignment

Hu's pioneering work [7] examined distributed code assignment for CDMA packet radio networks, including wireless ad hoc networks. Using the assumption that each node has a neighbor table updated by a network-layer routing protocol, Hu's approach transforms the code assignment problem into a graph theory problem. Hu defined four code assignment schemes, common code assignment (CCA), receiver-based code assignment (RCA), transmitter-based code assignment (TCA), and pair-wise code assignment (PCA) [7]. Because both TCA and PCA are NP-complete, fast heuristic algorithms are studied. Even though the solutions proposed in are sound, they have high time complexity and/or high communication overhead. Moreover, the schemes do not consider the case when the number of codes is limited and perfect assignment is not possible. Hu's scheme also assumes that each node has knowledge of its neighbors through a routing table. While this is applicable in most cases, assigning channels after the routing protocol converges could incur excessive delay and control overhead.

Garcia-Luna-Aceves and Raju [4] describe a distributed code assignment scheme that assigns distinct channels to a node and its two-hop neighbors. Each node sends out code assignment messages (CAM) that propagate up to two hops away from the node. If the number of codes available for assignment is at least $d(d-1)+2$, where d is the maximum degree, i.e., the maximum number of neighbors for any node, there will be no interference after the algorithm converges. However, this algorithm incurs high communication and computation overhead and has high time and storage complexity.

There are several recent proposals for routing protocols that are suitable for multi-hop multi-channel wireless mesh networks [11], [13], [15], [17]. The approach taken by most of these proposals is to combine routing with intelligent multi-channel assignment, such that channel utilization is maximized and the system performance can be substantially improved. However, because the routing protocols are emphasized, the effectiveness of the channel assignment schemes has not been studied [11], [13], [15]. Some channel assignment protocols have very high time complexity, i.e. $O(Kn^3 \log m + m^2)$ [17], where n is the total number of nodes in the network, m is the

TABLE II
COMPARISON OF CA-OLSR TO EXISTING ALGORITHMS

<i>Protocols</i>	<i>No. of Chann.</i>	<i>Comm. Complexity</i>	<i>Comp. Complexity</i>	<i>Storage Complexity</i>
Centralized greedy algorithm [7]	$d(d-1)+1$	N/A	$d^2 \cdot V $	$d^2 \cdot V $
Distributed channel assignment [4]	$d(d-1)+2$	N/A	d^2	d^2
Random scheme [11]	N/A	$O(1)$	$O(1)$	$O(1)$
Our CA-OLSR	$d_a(d_a-1)+1$	$O(1)$	$O(d_a^2)$	$O(1)$

total number of radio connections in the network, and K is the minimum number of neighbors for any node.

Since it has been proven that the distributed channel assignment problem is \mathcal{NP} -complete and, thus, computationally intractable, many heuristic algorithms have been proposed. However, most existing approaches require more channels than necessary and have high complexity [2], [4], [7], [17]. In addition, they also ignore harmful interference that may corrupt packets and cause retransmissions.

Table II compares several heuristic algorithms in terms of their communication, computation, and storage complexity. d is the maximum number of one-hop neighbors for any node, d_a is the maximum number of active one-hop neighbors for any node, n is the total number of active routes within a k -hop range of each other, and k is a constant associated with the interference range. For CA-OLSR, the channel assignment information is carried solely by routing messages. In addition, channel assignments for each neighbor node are saved along with each entry in the neighbor table. Therefore, there is very little communication overhead and storage overhead.

B. Related Work on Multi-Channel MAC Protocols

A significant body of prior work examines the benefits of utilizing multiple channels. Based on the way that a channel is selected, existing approaches fall into one of two categories, handshake-based channel selection and contention-based channel negotiation.

Handshake-based channel selection. Both dynamic channel assignment (DCA) [20] and the multi-channel CSMA MAC [8] utilize a handshake for channel selection. Just like the IEEE 802.11 standard, the handshake is realized by sending control messages between senders and receivers. Handshake-based channel selection is widely used in multi-channel MAC protocols and it works even when nodes are not synchronized. However, for the above two schemes, the negotiation of the channel-to-use is on a per-packet basis, which may incur high control overhead. Our proposed CA-OLSR scheme is invoked only when there is a channel conflict that needs to be resolved.

Contention-based channel negotiation Multi-channel MAC (MMAC) divides time into beacon intervals [14]. A small window is placed at the beginning of each beacon interval. During this window, all nodes transmit and receive on a common control channel and contend to select the desired data channel to use. Once a sender and a receiver successfully agree on a data channel, both of them switch to the data

channel for transmission at the end of the contention window. However, broadcasting is not supported in MMAC, which poses challenges for ad hoc routing protocols that rely on broadcasting for route queries. In addition, MMAC required nodes in the network to be synchronized, which may be hard to achieve in an ad hoc network.

According to the medium access control techniques utilized, multi-channel MAC protocols fall into one of the following two categories, CSMA/CA-based random access and frequency hopping.

CSMA/CA-based random access. Many of the multi-channel MAC protocols follow this approach because carrier sensing and handshake mechanisms are proven ways to avoid contention and collisions on wireless channels. Examples include DCA [20], MMAC [14], and a multi-channel CSMA MAC protocol [8]. Our proposed MC-MAC protocol falls into this category.

Channel hopping. Multi-channel MAC protocols that are based on logical channel hopping do not require carrier sensing and do not assign channels to individual nodes. Two such examples are receiver-initiated channel-hop with dual polling (RICH-DP) [18] and slotted seeded channel hopping (SSCH) [1].

According to the number of required transceivers or network interface cards (NICs), existing multi-channel MAC protocols can be divided into three categories, as described below.

Multiple transceivers. Some MAC protocols require more than one transceiver to take advantage of multiple channels. For example, DCA uses one transceiver to monitor a control channel and a second transceiver to transmit and receive on a data channel [20]. With added hardware complexity, DCA can utilize multiple channels with moderate control overhead.

One transmitter and multiple receivers. Jain, et al. assume that a node can receive packets on all channels simultaneously [8]. Thus, each node in a network is equipped with one transmitter and multiple receivers.

A single half-duplex transceiver. MMAC [14], MC-MAC [5], RICH-DP [18], and SSCH [1] all require that each node has only a single half-duplex transceiver. However, MMAC, RICH-DP, and SSCH require node synchronization to some extent. Our MC-MAC protocol does not require any node synchronization.

The multi-channel MAC schemes that require only a single transceiver can yield at least a factor 2 of improvement over single-channel MAC protocols [1], [14]. Our simulation results show an improvement up to a factor of 4. The decrease in

TABLE III
IMPORTANT DESIGN FEATURE OF EXISTING ALGORITHMS

<i>Protocols</i>	<i>Medium Access</i>	<i>Channel Selection</i>	<i>Hardware Requirement</i>
Dynamic Channel Assignment (DCA) [20]	CSMA/CA	Per packet	2 transceivers
Multi-channel MAC (MMAC) [14]	CSMA/CA	Per beacon interval	1 transceiver, synchronization required
Multi-channel CSMA [8]	CSMA/CA	Per packet	1 transmitter, multiple receivers
Receiver-Initiated Channel-Hop with Dual Polling (RICH-DP) [18]	Channel hopping	Hopping sequence	1 transceiver synchronization required
Slotted Seeded Channel Hopping (SSCH) [1]	Channel hopping	Hopping sequence	1 transceiver, synchronization required
Our Multi-Channel MAC (MC-MAC)	CSMA/CA	Per route change	1 transceiver

packet delay is even greater due to the significant reduction in the number of collisions and back-offs. However, because there is only one NIC available, a node can only operate on one channel at a time, which may result in inefficient channel utilization. A multi-channel MAC scheme that utilizes multiple NICs may yield better performance at the cost of higher hardware complexity. Increasing the number of transceivers per node may also cause excessive energy drain and consequent reduction of battery life in mobile devices. There is a fundamental tradeoff between the hardware complexity and the system performance. Because we combine channel assignment with routing protocols, the MC-MAC protocol only needs to implement the medium access control function, which greatly reduces the hardware complexity. Therefore, our proposed protocol has the minimum system requirement among all existing multi-channel MAC protocols.

We have classified existing multi-channel MAC protocols based on the 1) channel selection techniques, 2) medium access control schemes, and 3) hardware requirements. Table III summarizes some important features of existing multi-channel MAC protocols. There is no general rule as to which scheme is better than another. Simpler schemes with less hardware requirements are often easily made to be compatible with the IEEE 802.11 standard and are easy to implement, whereas complex schemes with greater hardware requirements often yield better performance.

VII. CONCLUSIONS

In this paper, we presented a combined proactive routing and multi-channel MAC protocol. The multi-channel MAC protocol is compatible with the IEEE 802.11 MAC and imposes the minimum system requirement among all existing multi-channel MAC protocols. The proposed CA-OLSR protocol exhibits significantly lower communication, computation, and storage complexity than existing channel assignment schemes. We proved the correctness of the CA-OLSR protocol. In addition, simulation results show that even with only a single half-duplex transceiver, CA-OLSR combined with MC-MAC can offer up to a factor of 4 improvement in throughput over the IEEE 802.11 MAC protocol.

REFERENCES

[1] P. Bahl, R. Chandra, and J. Dunagan, "SSCH: Slotted seeded channel hopping for capacity improvement in IEEE 802.11 ad-hoc wireless net-

works," in *Proc. IEEE/ACM the Tenth Annual International Conference on Mobile Computing and Networking (MobiCom)*, 2004, pp. 216–230.

[2] A. Bertossi and M. Bonuccelli, "Code assignments for hidden terminal interference avoidance in multihop packet radio networks," *IEEE/ACM Trans. on Networking*, vol. 3, no. 4, pp. 441–449, Aug. 1995.

[3] T. Clausen and P. J. eds., "Optimized Link State Routing Protocol," Oct. 2003, IETF RFC 3626.

[4] J. J. Garcia-Luna-Aceves and J. Raju, "Distributed assignment of codes for multihop packet-radio networks," in *Proc. MILCOM*, vol. 1, 1997, pp. 450–454.

[5] M. X. Gong and S. F. Midkiff, "A new multi-channel MAC protocol based a cross-layer approach," 2005, submitted for publication.

[6] M. X. Gong, S. F. Midkiff, and S. Mao, "Design principles for distributed channel assignment in wireless ad hoc networks," in *Proc. IEEE International Communications Conference (ICC)*, 2005.

[7] L. Hu, "Distributed code assignments for CDMA packet radio networks," *IEEE/ACM Trans. on Networking*, vol. 1, no. 6, pp. 668–677, Dec. 1993.

[8] N. Jain, S. R. Das, and A. Nasipuri, "A multichannel CSMA MAC protocol with receiver-based channel selection for multihop wireless networks," in *Proc. IEEE International Conference on Computer Communications and Networks (IC3N)*, 2001, pp. 432–439.

[9] M. Joa-Ng and I.-T. Lu, "Spread spectrum medium access protocol with collision avoidance in mobile ad-hoc wireless network," in *Proc. INFOCOM*, vol. 2, 1999, pp. 21–25.

[10] P. Kyasanur and N. Vaidya, "Capacity of multi-channel wireless networks: impact of number of channels and interfaces," University of Illinois at Urbana-Champaign, Tech. Rep., 2005.

[11] —, "Routing and interface assignment in multi-channel multi-interface wireless networks," in *Proc. IEEE Wireless Communications and Networking Conference (WCNC)*, 2005.

[12] J. Li, C. Blake, D. S. J. D. Coute, H. I. Lee, and R. Morris, "Capacity of ad hoc wireless networks," in *Proc. IEEE/ACM International Conference on Mobile Computing and Networking (MobiCom)*, 2001, pp. 61–69.

[13] A. Raniwala and T. Chiueh, "Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network," in *Proc. IEEE INFOCOM*, 2005.

[14] J. So and N. Vaidya, "Multi-channel MAC for ad hoc networks: Handling multi-channel hidden terminals using a single transceiver," in *Proc. ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, 2004, pp. 222–233.

[15] —, "Routing and channel assignment in multi-channel multi-hop wireless networks with single-NIC devices," University of Illinois at Urbana-Champaign, Tech. Rep., 2005.

[16] E. S. Sousa and J. A. Silvester, "Spreading code protocols for distributed spread-spectrum packet radio networks," *IEEE Trans. on Communications*, vol. 36, no. 3, pp. 272–281, March 1988.

[17] J. Tang, G. Xue, and W. Zhang, "Interference-aware topology control and QoS routing in multi-channel wireless mesh networks," in *Proc. the 6th ACM international symposium on Mobile ad hoc networking and computing (MobiHoc)*, May 2005, pp. 68–77.

[18] A. Tzamaloukas and J. J. Garcia-Luna-Aceves, "A receiver-initiated collision-avoidance protocol for multi-channel networks," in *Proc. IEEE INFOCOM*, 2001, pp. 189–198.

[19] VINT Group, *UCB/LBNL/VINT network simulator ns (version 2)*.

[20] 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 multihop mobile ad hoc networks," in *Proc. International Symposium on Parallel Architectures, Algorithms and Networks (ISPAN)*, 2000, pp. 232–237.