Asynchronous Multi-channel MAC Design with Difference Set based Hopping Sequences

Fen Hou, *Member, IEEE,* Lin X. Cai, *Member, IEEE,* Xuemin (Sherman) Shen, *Fellow, IEEE* and Jianwei Huang, *Member, IEEE* .

***Abstract*—Most existing multi-channel medium access control (MAC) protocols have at least one of the following performance bottlenecks: (1) global synchronization among users, (2) dedicated control channel for signaling exchange, (3) dedicated control phase for signaling exchange, and (4) complete knowledge of all users' channel selection strategies. In this paper, we first design a hopping sequence by combining multiple difference sets to assure a high rendezvous probability of users over multiple channels. Applying the hopping sequence to all users, we then propose a difference set based multi-channel MAC protocol (DSMMAC) to overcome the performance bottlenecks. As all users use the same sequence for frequency hopping and channel access, significant signaling overheads can be reduced. The proposed protocol achieves high system throughput and low access delay without the need for global synchronization or dedicated control channel/phase. Our analytical and simulation results show the proposed DSMMAC can achieve up to 100% improvement in system throughput and 150% reduction in channel access delay compared with an existing multi-channel MAC protocol.**

***Index Terms*—Difference set, medium access control, multi-channel, asynchronous.**

I. Introduction

Supporting successful concurrent transmissions of many close-by wireless links in the same channel is often difficult due to mutual interferences. Exploiting the multi-channel capability is an efficient approach to make concurrent transmissions possible. It is a general recognition that any pair of channels with at least 25 MHz frequency spacing are non-overlapping channels and can be used simultaneously without mutual interference [1]. Multi-channel communication systems have been widely adopted today to efficiently support concurrent transmissions in the frequency domain and achieve high network throughput (e.g., [2]–[5]). For example, three and twelve non-overlapping channels are specified in the widely deployed IEEE 802.11b and 802.11a wireless local area networks, respectively. In IEEE 802.11p, a seven channel structure is defined for intelligent transportation systems. The promising cognitive radio network is inherently a multi-channel system, where available "spectrum holes" are accessed opportunistically by the secondary users. A key design challenge for these systems is to have efficient medium access control (MAC) protocols that can coordinate the behavior of users and achieve a good system throughput.

F. Hou and J. Huang are with the Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong, China e-mail: ({fhou, jwhuang}@ie.cuhk.edu.hk). L.X. Cai and X. S. Shen are with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Ontario, Canada e-mail: ({lcai, xshen}@bbcr.uwaterloo.ca)

This work is supported by the General Research Funds (Project Number 412308 and 412509) established under the University Grant Committee of the Hong Kong Special Administrative Region, China.

Manuscript received August 11, 2010; revised December 07, 2010; accepted January 22, 2011.

In a multi-channel system, different users can use different channels to avoid excessive mutual interference. In order to perform the data transmission, the sender node and receiver node of a Source-Destination (S-D) pair must successfully negotiate with each other beforehand and agree on a common channel to operate. However, it is quite challenging to assure simple, robust, and efficient negotiations between S-D pairs, especially for a large scale wireless network. Most of the existing multi-channel MAC protocols try to reach this goal based on one or more of the following assumptions:

- Existence of a dedicated common control channel for signaling exchange.
- Existence of a dedicated control phase (time period) to exchange signaling information.
- Availability of global clock synchronization among all users.
- Every user's channel selection strategy (e.g., hopping sequence) is known by all other users in the network.

For a system with only a few channels available, using a dedicated control channel will significantly limit the available resource and thus the network performance. Also, heavy control traffic may make the control channel the performance bottleneck for the whole network. Furthermore, it is not always possible to assign a dedicated control channel if prior coordination among users is not possible (e.g., in cognitive radio networks). The global synchronization can help to define the dedicated control phase as in [16], but it is often difficult to achieve in large distributed networks and will also introduce additional implementation complexity.

In order to achieve the efficient S-D negotiation as well as avoiding the global time synchronization and dedicated control channel, we introduce a difference set based frequency hopping in this paper. Difference set has been widely used in the design of power-saving protocol to reduce the wake-up duty cycle while maintaining a probability of communicating with neighboring nodes [6] [7]. Different from sleep scheduling algorithms which aim at minimizing the energy consumption, our objective is to achieve high channel utilization and network throughput in a multiple channel system. Recently, [8] proposes to apply quorum based channel hopping (QCH) in a synchronous multi-channel system for control channel establishment. An asynchronous QCH scheme for a two-channel system is also introduced. To the best of our knowledge, asynchronous distributed MAC design in a system with more than 2 channels is still an open issue. Moreover, with more channels, how to apply *difference set* based sequence design to assure efficient, robust, and fair channel access among multiple users become more challenging.

In this paper, we propose a novel *difference set* based asynchronous multi-channel MAC protocol (DSMMAC) for distributed wireless network without control channels. The unique rotation closure property of difference sets enables any two nodes to communicate (i.e., meet or rendezvous) with a non-zero probability without a global synchronization or a dedicated control channel (the detailed discussion on the rendezvous probability is given in Section III). By combining multiple disjoint difference sets, we construct a hopping sequence suitable for multi-channel networks. Besides the performance metric of time to rendezvous, we are also interested in network throughput, channel utilization, channel access fairness, and power consumption, etc. Specifically, the main advantages of the proposed DSMMAC protocol include:

1) *Asynchronization*: DSMMAC does not require global synchronization among users, and thus is suitable for implementation in distributed networks.
2) *Multiple parallel rendezvous*: Multiple source-destination pairs can rendezvous and communicate over different channels simultaneously without a centralized coordination. This improves the channel utilization and system performance.
3) *Less signaling overhead*: All users deploy the same hopping sequence in DSMMAC, which eliminates the need for exchanging hopping sequence information and reduces the signaling overhead.
4) *No dedicated control channel*: Since DSMMAC requires no dedicated common control channel, it can fully utilize all available channels for data transmissions.

The remainder of the paper is organized as follows. Section II describes the network model. The difference set based hopping sequence is discussed in Section III, followed by the proposed DSMMAC protocol in Section IV. An analytical model is developed in Section V. Simulation results are presented in Section VI. Section VII presents the related work. We finally conclude in Section VIII.

II. Network model

We consider a multi-channel network consisting of sets $\mathcal{C} = \{c_1, \cdots, c_L\}$ of L channels and $\mathcal{N} = \{n_1, \cdots, n_{2M}\}$ of $2M$ nodes. An example is shown in Figure 1, where $L = 4, M = 4$. Each node represents an end-user which can be a mobile phone, a mobile PDA, a laptop, or a sensor. Two nodes that want to communicate with each other are called a source-destination (S-D) pair. Each node is equipped with one tunable half-duplex radio transceiver which can switch between different channels. Whether or not an S-D pair can communicate depends on their selected working channels. As shown in Figure 1, the nodes n_3 and n_8 turn their working channels to c_4 and c_3, respectively. The destination node n_8 can not listen the information transmitted by the source node n_3 over the channel c_4. Therefore, the S-D pair $n_3 - n_8$ can not communicate with each other. An S-D pair can communicate only when both the source and destination nodes turn to the same channel simultaneously (i.e., they meet or rendezvous over a channel). As shown in Figure 1, the S-D pair $n_4 - n_6$ select the same channel c_2. They can conduct the data transmission after the successful negotiation over the channel c_2.

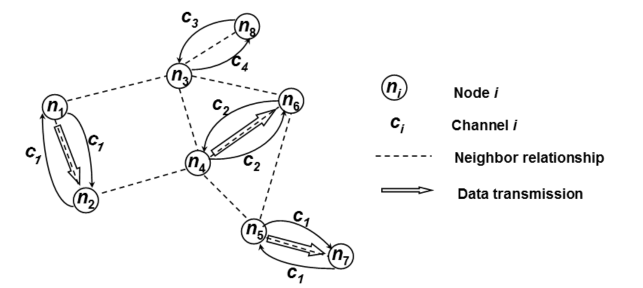


Fig. 1. A generic multi-channel wireless network with 8 nodes and 4 channels.

We adopt the carrier sensing multiple access (CSMA) with RTS (Ready to Send)/CTS(Clear to Send) access control mechanism in our system model. All nodes have the same transmission and interference ranges. Multiple S-D pairs can communicate simultaneously over the same channel if the interferers do not cause harmful interference to the ongoing transmissions, i.e., they are outside the interference range of the transmitting S-D pairs. As shown in Figure 1, two S-D pairs (i.e., n_1 - n_2, and n_5 - n_7) can communicate simultaneously over channel c_1 since these two pairs are far away and do not interfere with each other. In a multi-channel system, multiple S-D pairs can communicate concurrently over different channels thanks to the channel orthogonality.

To make an S-D pair rendezvous, the source and destination hop over channels independently based on a pre-defined hopping sequence. Detailed hopping sequence and multi-channel MAC design will be presented in Sections III and IV.

III. Difference-Set Based Hopping Sequence

The key idea behind our proposed multi-channel MAC protocol is to design the hopping sequence based on the concept of *difference set*. Different from the use of difference set in the design of power-saving sleep scheduling algorithms [9], where the main objective is to minimize a node's wake-up duty cycle, our objective is to maximize the rendezvous probability of S-D pairs. Moreover, we need to consider the design in both frequency and time domains to guarantee a successful rendezvous. In the following, we first review the properties of difference set [10] [11]. Then we present the design of difference set based hopping sequence for a multi-channel system.

A. Definitions and Properties of Difference Sets

Definition 1 (Difference Set) : A difference set is defined by three elements: the set cycle v, the set size k, and the time number λ. A (v, k, λ) difference set includes k integers, denoted by $\mathcal{D} = \{d_1, \cdots, d_k\}$. For each value r that is smaller than v (i.e., $r \in \{1, \cdots, v-1\}$), we can find exact λ ordered pairs of (d_i, d_j) in the set such that the difference is r (i.e., $r = (d_i - d_j) \mod v$, where $d_i, d_j \in \mathcal{D}$).

For example, the set $\{1, 2, 4\}$ is a $(7, 3, 1)$ difference set. There are 3 elements in this set with the cycle of 7. For any integer $r \in \{1, 2, \cdots, 6\}$, we can find exactly one ordered pair of elements in the set which gives the difference of r . For instance, the ordered pair (1,2) gives a difference of 6 (i.e., $6 = (1-2) \mod 7$); The ordered pair (2,1) gives a difference of 1, etc.

Definition 2 (Complementary set of a difference set) : Let $\mathcal{D} = \{d_1, \cdots, d_k\}$ be a difference set with a cycle of v. The complementary set of $\mathcal{D}$ is defined with respect to the set of $\mathcal{Z} = \{1, 2, \cdots, v\}$, i.e.,

$$\overline{\mathcal{D}} = \{l : l \in \mathcal{Z} \setminus \mathcal{D}\}$$

Take the set $\{1, 2, 4\}$ as an example. It is a difference set with the cycle of 7. Therefore, its complementary set is $\{3, 5, 6, 7\}$.

Definition 3 (Shift set of a difference set) : Let $\mathcal{D} = \{d_1, \cdots, d_k\}$ be a difference set with the cycle of v. Its μ^{th} shift set is defined as $\mathcal{D}^\mu = \{d_1 + \mu, \cdots, (d_k + \mu)\ (mod\ v)\},\ \ \mu \in \{1, \cdots, v\}$.

The shift set of a difference set is generated by conducting a shift operation with the mod of v. For instance, the 1^{st} shift set of the difference set $\{1, 2, 4\}$ is $\{2, 3, 5\}$.

Property 1: The complementary set of a difference set is a difference set - If $\mathcal{D}$ is a difference set with the cycle of v, then its complementary set $\overline{\mathcal{D}}$ is also a difference set with the cycle of v.

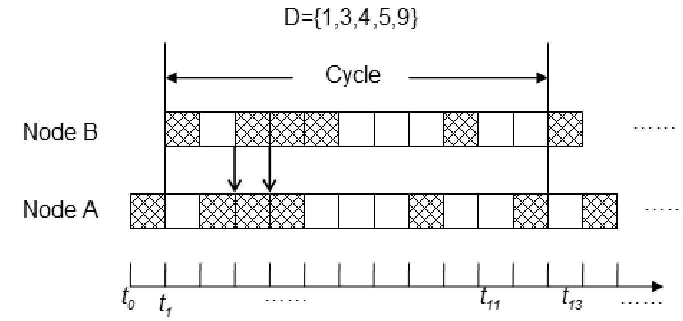
We can use this property to generate multiple disjoint difference sets with the same cycle.

Property 2: Rotation closure property - A (v, k, λ) difference set $\mathcal{D}$ and any of its shift sets have λ overlapping elements in a cycle of v.

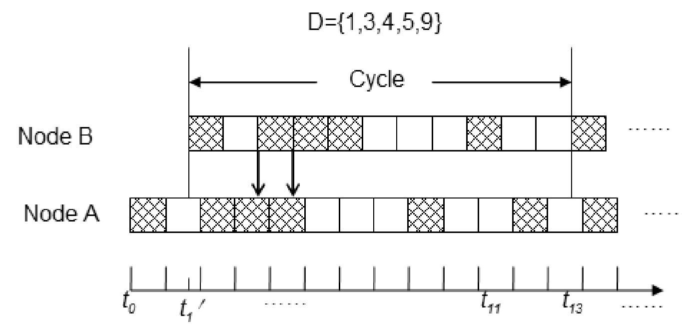
When applying the rotation closure property to the design of hopping sequence, two nodes using the same difference set in either synchronous or asynchronous fashions will have a rendezvous probability λ/v without prior coordination. Examples are shown in Figure 2 and elaborated further as follows.

Figure 2 illustrates the rotation closure property in a single channel case (e.g., Channel c_1). Nodes A and B follow the same hopping sequence generated from a $(11, 5, 2)$ difference set $\mathcal{D} = \{1, 3, 4, 5, 9\}$. In a cycle of 11 time slots, nodes hop over channel c_1 in their 1, 3, 4, 5, and 9 time slots depicted as solid rectangles. An empty rectangle represents an inactive time slot in which users switch off their transceivers.

- Synchronization Case: As shown in Figure 2(a), nodes A and B have aligned time slot boundaries. They start to hop to channel c_1 at the time t_0 and t_1, respectively, based on the same hopping sequence but with different time shifts. Let us consider a cycle of 11 time slots. We can see that two nodes rendezvous on the channel c_1 twice (the beginnings of node B's time slot 3 and 4). The long-term rendezvous probability is $2/11$.
- Asynchronization Case: As shown in Figure 2(b), the time slot boundaries are not aligned. nodes A and B join the network and hop to channel c_1 at the time t_0 and t'_1, respectively. Within a cycle of 11 time slots, two nodes rendezvous on the channel c_1 twice (at the beginning of node B's 3 and 4 time slots). The long-term rendezvous probability is still $2/11$.



(a) Aligned boundaries of time slots.



(b) Nonaligned boundaries of time slots.

Fig. 2. Illustration of a rotation closure property for the difference set $D = \{1, 3, 4, 5, 9\}$.

The above example illustrates that the rendezvous probability λ/v is independent of synchronization.

B. The design of difference set based hopping sequence

A hopping sequence determines which channel a user should access during each time slot of a cycle. In the proposed DSMMAC protocol, all users use the same frequency hopping sequence so that users do not need to exchange hopping sequences with each other. The hopping sequence is designed with multiple difference sets in order to support communications over multiple channels. In addition, it is desirable to achieve a high rendezvous probability of each S-D pair so that they have a higher chance to successfully communicate with each other. Moreover, we will try to support many concurrent transmissions to exploit the capability of multi-channel networks. The main design steps include:

Step 1: Select multiple difference sets - To guarantee successful node rendezvous in both time and frequency domains, we need to combine multiple difference sets with the *same* cycle to generate a hopping sequence covering all channels. That is, for a network with L available channels, we will choose L difference sets with the same cycle v. In other words, we combine L difference sets for channel access in the frequency domain, and each difference set has a cycle v that assures the rendezvous probability λ/v over each channel. We propose two criteria for the sequence design as follows.

Criterion 1: High rendezvous probability - We select difference sets with high rendezvous probabilities (i.e., large values of λ/v) to assure that users are very likely to meet with each other over each channel.

Criterion 2: Empty intersection - Any two chosen difference sets (denoted as $\mathcal{D}_i$ and $\mathcal{D}_j$ $(i \neq j)$) should satisfy

$$\mathcal{D}_i \bigcap \mathcal{D}_j = \emptyset, \quad \forall\ i, j \in \{1, 2, \cdots L\}.$$

In a multi-channel network, two users can communicate when they hop on the same channel at the same time (i.e., meet in both frequency and time domains). Criterion 1 makes sure that nodes meet with high probabilities. Criterion 2 makes sure that there is no ambiguity when assigning channels to time slots in a hopping sequence. A numerical example later on will clearly illustrate this point.

Step 2: Match the frequency channels with the selected difference sets - Let $\mathcal{D}_1, \mathcal{D}_2, \cdots$, and $\mathcal{D}_L$ be L difference sets selected for a network with L available channels (denoted as $c_1, c_2, \cdots, c_L$). We will associate one channel to each difference set, and allow access of this channel only during the time slots of the corresponding difference set. For instance, $D_1 = \{1, 3, 6\}$ means that the node will only access channel c_1 during time slots 1, 3, 6 of a cycle.

Step 3: Randomly allocate the frequency channels to the remaining unassigned time slots - It is possible that some time slots remain unassigned after Step 2 if the total number of elements of all L different sets is smaller than v. In other words. Let $\Psi = \{\mathcal{D}_1, \mathcal{D}_2, \cdots, \mathcal{D}_L\}$ be the set of selected difference sets, and $\overline{\Psi}$ be the complementary set of Ψ in $\{1, 2, ...v\}$. If $\overline{\Psi} \neq \emptyset$, we randomly assign channels to the time slots in $\overline{\Psi}$ for easy implementation.

As a simple example, we first elaborate the hopping sequence design for a network with $L = 2$ channels (denoted as c_1 and c_2). In this case, two difference sets with the same cycle are needed. By applying Property 1 of the difference set, we choose difference set $\{1, 2, 4\}$ and its complementary difference set $\{3, 5, 6, 7\}$ (i.e., $\mathcal{D}_1 = \{1, 2, 4\}$, and $\mathcal{D}_2 = \{3, 5, 6, 7\}$). Both of them have the same cycle of 7 time slots. The rendezvous probabilities of the two difference sets are $1/7$ and $2/7$, respectively. Moreover, these two difference sets do not intersect with each other (i.e., Criterion 2 in design Step 1). We allocate the frequency hopping channels c_1 and c_2 based on these two difference sets as follows. Channel c_1 is allocated to the time slots of $\mathcal{D}_1$ (i.e., time slots 1, 2, and 4), and channel c_2 is allocated to the time slots of $\mathcal{D}_2$ (i.e., time slots 3, 5, 6, and 7). The resulting hopping sequence is H=$[c_1\ \ c_1\ \ c_2\ \ c_1\ \ c_2\ \ c_2\ \ c_2]$ shown in Figure 3.

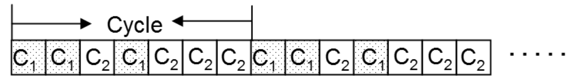


Fig. 3. A hopping sequence for two channels based on two difference sets.

As a more complicated example, we consider the hopping sequence design for 8 channels (denoted as c_1 through c_8). We first choose 8 difference sets as follows

$$\Psi = \{\mathcal{D}_1,\ \mathcal{D}_2,\ \mathcal{D}_3,\ \mathcal{D}_4,\ \mathcal{D}_5,\ \mathcal{D}_6,\ \mathcal{D}_7,\ \mathcal{D}_8\}$$
$$= \{\{2\ 3\ 5\ 9\ 17\ 33\ 38\ 56\ 65\},\ \{4\ 7\ 13\ 20\ 24\ 25\ 39\ 47\ 49\},\ \{6\ 8\ 11\ 15\ 21\ 29\ 40\ 41\ 57\},\ \{10\ 19\ 37\ 42\ 58\ 66\ 70\ 72\ 73\},\ \{12\ 16\ 22\ 23\ 31\ 43\ 45\ 48\ 61\},\ \{14\ 27\ 30\ 32\ 44\ 52\ 53\ 59\ 63\},\ \{18\ 34\ 35\ 46\ 54\ 60\ 64\ 67\ 69\},\ \{26\ 28\ 36\ 50\ 51\ 55\ 62\ 68\ 71\}\}.$$

These 8 difference sets have the same cycle (i.e., $v = 73$ time slots) and empty intersections (i.e., Criterion 2). The frequency hopping channels are allocated based on the difference sets. For instance, at the time slots corresponding to $\mathcal{D}_1$ (i.e., the time slots 2, 3, 5, 9, 17, 33, 38, 56, and 65), the frequency hopping channel is c_1. After allocating channels based on all 8 difference sets, we find that time slot 1 remains unassigned (i.e., $\overline{\Psi} = \{1\}$). Therefore, we randomly assign one frequency channel to the time slot 1. Without loss of generality, we allocate channel c_1 to the time slot 1. The final hopping sequence is H=$[c_1\ c_1\ c_1\ c_2\ c_1\ c_3\ c_2\ c_3\ c_1\ c_4\ c_3\ c_5\ c_2\ c_6\ c_3\ c_5\ c_1\ c_7\ c_4\ c_2\ c_3\ c_5\ c_5\ c_2\ c_2\ c_8\ c_6\ c_8\ c_3\ c_6\ c_5\ c_6\ c_1\ c_7\ c_7\ c_8 c_4\ c_1\ c_2\ c_3\ c_3\ c_4\ c_5\ c_6\ c_5\ c_7\ c_2\ c_5\ c_2\ c_8\ c_8\ c_6\ c_6\ c_7\ c_8\ c_1\ c_3\ c_4\ c_6\ c_7\ c_5\ c_8\ c_6\ c_7\ c_1\ c_4\ c_7\ c_8\ c_7\ c_4\ c_8\ c_4\ c_4]$.

IV. Difference-set based Multi-channel MAC Design

The key idea of our proposed DSMMAC (Difference Set based Multi-channel MAC) is to design a smart difference-set based hopping sequence. Then, all nodes use this sequence for channel hopping and channel access as follows. When a source node joins the network, it randomly selects an initial channel and attempts to access the channel by starting a handshake process. If the source node successfully exchanges handshake information with its intended destination node, the sender-destination pair stops hopping and starts data transmissions over this channel. Otherwise, the source node selects the next hopping channel in the following time slot according to the pre-designed hopping sequence. The node keeps channel hopping until it successfully shakes hands with its destination.

The proposed DSMMAC has several advantages. First, all nodes use the same hopping sequence that is pre-designed (e.g., by the manufacturer or the service provider), thus there is no need for further information exchange, and no central controller or centralized allocation is needed. Second, the unique rotation closure property of difference set ensures that a source node is able to successfully meet its destination with a certain probability even without synchronization. Third, although all nodes use a common hopping sequence, nodes join the network at different time instants and initiate channel accesses over different channels, which helps to achieve multiple parallel rendezvous and improve the channel utilization. The detailed algorithms of the source and destination nodes are presented in Algorithm 1 and Algorithm 2, respectively.

When a node joins the network, it randomly selects a hopping channel and starts channel sensing. If the channel is sensed busy, which implies that the current channel is occupied by other S-D pairs, the source node will switch to the next hopping channel based on the hopping sequence and resume the channel sensing at the next time slot. If the channel is sensed idle for a DIFS (DCF InterFrame Space) interval, implying currently there is no ongoing transmissions over the channel, the source node initiates a RTS transmission. If no CTS is received after an SIFS (Short InterFrame Space) interval, it continues channel hopping. Notice that it is also possible that the transmitted RTS/CTS are lost in the error-prone wireless channel or due to hidden terminal problems in a multi-hop network. If the channel condition between an S-D pair is poor, it is not desirable to start data transmission in the first place. In addition, with multiple channels operating in parallel, the collision probability on any one of these channels is greatly reduced. Therefore, when a source node sends a RTS but receives no CTS after an SIFS, most likely the corresponding destination does not access the same channel at this moment. Thus, the source node should switch to a different channel. If a CTS is received after an SIFS, which means the corresponding destination also accesses the channel at this time. After the successful handshake, the source node starts data transmissions to its destination.

To allow multiple S-D pairs to fairly share the wireless medium, users release the channel when a predefined time duration T_{max} is reached or its data buffer becomes empty. After releasing the current channel, a node resumes channel hopping based on the hopping sequence until a successful handshake. Because of the rotation closure property of difference sets, any two users can meet with each other in one channel with a certain probability, regardless of the initial choice of hopping channels they select and channel synchronization.

Algorithm 1 Source node

```
Randomly select a hopping channel;
while There is data to send do
    Sense the channel;
    if Channel is idle for a DIFS then
        Send a RTS;
        if A CTS is received within a SIFS then
            while Channel occupation time ≤ T_max and data buffer
            is not empty do
                Transmit data over the channel;
            end while
            Release the channel;
        end if
    end if
    Hop to the next channel based on the designed hopping
    sequence;
end while
```

Algorithm 2 Destination node

```
Randomly select a hopping channel;
Sense the channel;
if Channel is idle then
    Wait a time slot for a RTS;
    if Receive a RTS targeted to it then
        Respond with a CTS;
        if Receive data within an SIFS then
            repeat
                Receive data
            until Data transmissions complete
        end if
    end if
end if
Hop to the next channel based on the designed hopping sequence;
```

As shown in Algorithm 2, the operation of the destination node is similar. The main difference is that a destination node will wait for receiving a potential RTS for a time slot and respond with CTS.

In the proposed MAC protocol, frequency hopping is only used to find the rendezvous channel. After successful rendezvous occurs between an S-D pair, they stop the frequency hopping and conduct their data transmission over the common channel. In other words, the data transmission is not based on frequency hopping technique, but over a single channel. Therefore, the proposed protocol has much less signalling overhead than the traditional frequency hopping technique-based MAC protocol (i.e. Frequency Hopping Spread Spectrum in IEEE 802.11). In the latter case, data is sent over a sequence of the hopping channels corresponding to an agreed hopping *pattern* between an S-D pair.

V. Performance Analysis

We analyze the performance of the proposed DSMMAC in terms of network throughput and channel utilization. Main notations used in the paper are listed in TABLE I.

We divide time into small "virtual" slots, denoted as time slot t $(t = 1, 2, \cdots)$. Notice that the time slots are introduced here for analytical convenience. In practical implementations, each node keeps a local time-slotted system and the boundaries of time slots are not necessarily synchronized across users. The only requirement is that the length of a time slot is the same for all users. We observe the system at each time slot and assume that system state transition occurs only at the beginning of each time slot. Denote a system state, e.g., system state i, as $\boldsymbol{s}_i \in \boldsymbol{s}$, which consists of a set of channels being used for data transmissions at this time slot.

The sequence of observed system states then forms a discrete Markov chain. Without loss of generality, in system state i, $\boldsymbol{s}_i = \{c_i^1, c_i^2, \cdots, c_i^I\}$, where $c_i^1, \cdots, c_i^I$ represent the channels involving in the data transmissions at this state. For instance, $\boldsymbol{s}_{10} = \{c_1, c_2, c_8\}$ means that channels c_1, c_2 and c_8 are being used for active data transmission and this system state is labeled as state 10 in the Markov chain. The labeling of system state is simply a mapping from the system state to an integer. For instance, for a network with two available channels, the possible system states are $\emptyset$, $\{c_1\}$, $\{c_2\}$, $\{c_1, c_2\}$.

TABLE I
TABLE OF NOTATIONS.

L	the total number of channels in the system
c_i	The i^{th} channel in the system
M	the total number of S-D pairs
H	Hopping sequence
h_i	The i^{th} hopping channel in H
$\boldsymbol{s}_i$	system state i
ϕ^f_{ij}	the set of channels that finish data transmission at the system transition from the state i to j
ϕ^b_{ij}	the set of channels that begin data transmission at the system transition from the state i to j
T	Mean of each data transmission (in unit of time slot)
q	Probability of finishing data transmissions at a time slot
$Pr[F_{ij}]$	Probability of finishing the transmissions over all channels in ϕ^f_{ij}
$Pr[B_{ij}]$	Probability of beginning the transmissions over all channels in ϕ^b_{ij}
$Pr[W^m_i]$	Probability that the m^{th} S-D pair fails to negotiate given system state i
$Pr[B^{m,l}_i]$	Probability that the m^{th} S-D pair begins data transmission over channel l given system state i
μ	Channel utilization
r	Data transmission rate
Th	System throughput
Th_m	Achieved throughput of the m^{th} S-D pair

These four system states can be labeled as the states 1, 2, 3, and 4, respectively.

The one-step transition probability from the state i to j is defined as

$$p_{ij} = Pr[s(t+1) = \boldsymbol{s}_j | s(t) = \boldsymbol{s}_i] \tag{1}$$

where $\boldsymbol{s}_i = \{c^1_i, \cdots, c^I_i\}$ and $\boldsymbol{s}_j = \{c^1_j, \cdots, c^J_j\}$. Here I and J represent the total number of active data transmission channels at states i and j, respectively. The value of p_{ij} is independent of the time index t.

Additional notations are needed to derive the values of transition probability p_{ij}. Let ϕ^f_{ij} and ϕ^b_{ij} denote the sets of channels finishing and beginning data transmissions at the state transition from i to j, respectively. Therefore, two events occur during this state transition: (1) active S-D pairs over channels in ϕ^f_{ij} finish their transmissions; and (2) S-D pairs begin new transmissions over channels in set ϕ^b_{ij}. For instance, given that $\boldsymbol{s}_i = \{c_1, c_3, c_4\}$, and $\boldsymbol{s}_j = \{c_1, c_5, c_8\}$, we have $\phi^f_{ij} = \{c_3, c_4\}$, and $\phi^b_{ij} = \{c_5, c_8\}$. Let $Pr[F_{ij}]$ be the probability that data transmissions over channels in $|\phi^f_{ij}|$ finish, and $Pr[B_{ij}]$ be the probability that new data transmissions over channels in $|\phi^b_{ij}|$ begin. The one step transition probability from the state i to j is

$$p_{ij} = Pr[F_{ij}]Pr[B_{ij}]. \tag{2}$$

A. Derivation of $Pr[F_{ij}]$

Let random variable X represent the duration of a data transmission. To simplify the analysis, in this paper we assume that durations of all data transmissions follow the same exponential distribution with mean T. To formulate the state transition into a discrete Markov model, we convert the continuous random variable X to a discrete random variable $Y = \lceil X/T_{slot} \rceil$, where T_{slot} is the duration of a time slot and $\lceil \cdot \rceil$ represents the largest integer less than or equal to the argument. Since X is an exponential random variable with mean T , then X/T_{slot} also follows an exponential distribution with mean $\alpha = T/T_{slot}$, the discrete random variable follows geometric distribution with parameter $q = (1 - e^{-1/\alpha})$. In other words, we observe the system at the beginning of each time slot, each data transmission may finish or continue with probability $q = (1 - e^{-1/\alpha})$ and $1 - q = e^{-1/\alpha}$, respectively. The accuracy of this approximation will be demonstrated in Section VI.

During one-step transition from the state i to j, the data transmissions over channels in ϕ^f_{ij} finish while the data transmissions over other channels in $\boldsymbol{s}_i$ are still active. The probability of this event is

$$Pr[F_{ij}] = q^{|\phi^f_{ij}|}(1-q)^{|\boldsymbol{s}_i| - |\phi^f_{ij}|}, \tag{3}$$

where $|\cdot|$ represents the cardinality of a set.

B. Derivation of $Pr[B_{ij}]$

A data transmission starts when an S-D pair successfully rendezvouses over one channel. In a synchronous network, the transmission boundaries of RTS signalling are aligned, which leads to a non-zero probability of concurrent transmissions over the same channel (i.e., collision). In this paper, we consider that S-D pairings are fixed and focus on the asynchronous case where the probability that two or more sources attempting to transmit over the same channel simultaneously is negligible. In this case, the probability of a successful rendezvous only depends on the hopping sequence and the total number of S-D pairs attempting to access the channel. Let $Pr[B^{m,l}_i]$ be the probability that the m^{th} S-D pair begins data transmission over channel c_l at the next time slot, given that the system is in state i at the current time slot. It equals to the probability of successful rendezvous over channel c_l for an S-D pair.

The value of $Pr[B^{m,l}_i]$ depends on how the source and destination nodes of the m^{th} S-D pair select their initial hopping channels. Let η_s and η_d be the indices of initial hopping channels selected by the source and destination, respectively. For instance, consider a hopping sequence $H = [h_1, \cdots, h_g, \cdots, h_v]$, where v is the cycle of the hopping sequence and $h_g \in \{c_1, c_2, \cdots, c_L\}$ represents the g^{th} hopping channel in the hopping sequence H. $\eta_s = 2$ means that the source node selects h_2 as its initial hopping channel. Thus, the probability of successful rendezvous over channel c_l is

$$Pr[B_i^{m,l}] = \sum_{\gamma_1=1}^{v}\sum_{\gamma_2=1}^{v} Pr[\eta_s=\gamma_1]Pr[\eta_d=\gamma_2]Pr[B_i^{m,l}|\eta_s,\eta_d]. \tag{4}$$

Given η_s and η_d, the conditional probability $Pr[B_i^{m,l}|\eta_s,\eta_d]$ depends on the hopping sequence H. As an example, assume $\eta_s = 5$, $\eta_d = 1$, and $H = [c_1, c_1, c_2, c_1, c_2, c_2, c_2]$. The source and destination nodes choose the channels $h_5 = c_2$ and $h_1 = c_1$ as their initial hopping channels respectively. Therefore, they will meet at the third time slot in which both of them hop to channel c_2. The conditional probability is $Pr[B_i^{m,2}|\eta_s = 5, \eta_d = 1] = 1/3$.

Let $\mathcal{S}_i$ and $\overline{\mathcal{S}}_i$ be the set of active channels at the system state $\boldsymbol{s}_i$ and its complementary set in $\{c_1, c_2, \cdots, c_L\}$, respectively. Thus, the probability that the m^{th} S-D pair fails to negotiate for a data transmission, given the system to be at state $\boldsymbol{s}_i$, is

$$Pr[W_i^m] = 1 - \sum_{l\in\overline{\mathcal{S}}_i} Pr[B_i^{m,l}] \tag{5}$$

Let Ω represent the set of S-D pairs attempting to access the channels during the transition from the state i to j. We have $\Omega = M - |\boldsymbol{s}_i|$, where M is the total number of S-D pairs, and $|\boldsymbol{s}_i|$ is the number of active S-D pairs at the state i. When the system transits to the state j, there are $|\phi_{ij}^b|$ S-D pairs successfully negotiate for their data transmissions over channels in ϕ_{ij}^b. There exist $\binom{M-|\boldsymbol{s}_i|}{|\phi_{ij}^b|}$ possible combinations of selecting $|\phi_{ij}^b|$ out of $M - |\boldsymbol{s}_i|$ S-D pairs. Let Ω_φ represent the set of S-D pairs corresponding to the φ^{th} combination, and $\overline{\Omega}_\varphi$ be the complementary set of Ω_φ in Ω. Therefore, $\prod_{m\in\Omega_\varphi, m'\in\overline{\Omega}_\varphi, l\in\phi_{ij}^b} Pr[B_i^{m,l}]Pr[W_i^{m'}]$ is the probability that S-D pairs in Ω_φ successfully negotiate over the channels in ϕ_{ij}^b and S-D pairs in $\overline{\Omega}_\varphi$ fail their negotiations. By summing over all possible combinations from $\varphi = 1$ through $\binom{M-|\boldsymbol{s}_i|}{|\phi_{ij}^b|}$, we obtain the probability that channels in ϕ_{ij}^b begin data transmissions as

$$Pr[B_{ij}] = \sum_{\varphi=1}^{\binom{M-|\boldsymbol{s}_i|}{|\phi_{ij}^b|}} \prod_{m\in\Omega_\varphi, m'\in\overline{\Omega}_\varphi, l\in\phi_{ij}^b} Pr[B_i^{m,l}]Pr[W_i^{m'}]. \tag{6}$$

As an illustrative example, consider a total number of $L = 2$ channels denoted as c_1 and c_2. The number of S-D pairs is $M = 3$. The system states in two consecutive time slots are $\boldsymbol{s}(t) = \boldsymbol{s}_i = \{c_1\}$ and $\boldsymbol{s}(t+1) = \boldsymbol{s}_j = \{c_1, c_2\}$, respectively. Therefore, we have $\phi_{ij}^f = \emptyset$, $\phi_{ij}^b = \{c_2\}$, and $\Omega = \{1, 2\}$. Thus, $Pr[B_{ij}]$ is obtained as

$$\begin{aligned} Pr[B_{ij}] &= \sum_{\varphi=1}^{\binom{2}{1}} \prod_{m\in\Omega_\varphi, m'\in\overline{\Omega}_\varphi, l\in\phi_{ij}^b} Pr[B_i^{m,l}]Pr[W_i^{m'}] \\ &= \prod_{m\in\{1\}, m'\in\{3\}, l\in\{c_2\}} Pr[B_i^{m,l}]Pr[W_i^{m'}] \\ &+ \prod_{m\in\{3\}, m'\in\{1\}, l\in\{c_2\}} Pr[B_i^{m,l}]Pr[W_i^{m'}] \\ &= Pr[B_i^{1,2}]Pr[W_i^3] + Pr[B_i^{3,2}]Pr[W_i^1]. \end{aligned} \tag{7}$$

C. System throughput and channel utilization

Based on the derived probabilities of finishing and beginning data transmissions, the one-step transition probability matrix is

$$\begin{aligned} \mathbf{P} &= [p_{ij}]_{N\times N}; \quad i,j \in \Theta \\ p_{ij} &= Pr[F_{ij}]Pr[B_{ij}] \end{aligned} \tag{8}$$

where Θ represents the state space of the Markov chain, and $N = |\Theta|$ is the total number of states in Θ.

Let π_i denote the steady-state probability for the system staying at the state i. The system steady-state probability, $\mathbf{\Pi} = [\pi_1, \pi_2, ..., \pi_N]$ can be calculated as

$$\mathbf{\Pi} = \boldsymbol{Q}[(\boldsymbol{I} - \boldsymbol{P})\mathbf{\Lambda} + \boldsymbol{U}]^{-1}, \tag{9}$$

where $\boldsymbol{Q}$ is a 1-by-N zero vector except that the last element is one, $\boldsymbol{I}$ is a N-by-N identity matrix, $\boldsymbol{P}$ is the one-step transition matrix, $\mathbf{\Lambda}$ is a N-by-N matrix with the first $(N-1)$ elements of the diagonal set equal to 1 and other elements are zero, $\boldsymbol{U}$ is N-by-N zero matrix except that all elements in the last column are ones. The notation $[\cdot]^{-1}$ represents the matrix inverse. Thus, the average number of channels used for the data transmissions is

$$\overline{L} = \sum_{i\in\Theta} n_i\pi_i \tag{10}$$

where n_i is the number of channels used for the data transmission when the system is the state i.

The average system throughput is

$$Th = r.\sum_{i\in\Theta} n_i\pi_i \tag{11}$$

where r is the average data transmission rate of an S-D pair, and is assumed to be the same for all S-D pairs.

Define the channel utilization as the ratio of the channels involved in the data transmissions to the total number of channels in the system, i.e.,

$$\mu = \left(\sum_{i\in\Theta} n_i\pi_i\right)/L. \tag{12}$$

TABLE II
TABLE OF THE MAIN SIMULATION PARAMETERS

Parameter	Value	Parameter	Value
Channel number L	2, 8	SIFS	16 μs
S-D pair number M	$1-20$	DIFS	34 μs
PHY preamble	192 bits	RTS	20 bytes
Channel switch time	40 μs	CTS	14 bytes
Slot duration	418 μs	ACK	14 bytes
Transmission/interference range	100 m	RTS/CTS rate	2 Mbps
Mean of transmission duration T	$5-20$(slots)	Data rate r	54 Mbps

VI. NUMERICAL RESULTS

We evaluate the performance of the proposed DSMMAC in terms of system throughput, channel utilization, fairness, and power consumption overhead using an event-driven C simulator. The number of S-D pairs ranges from 1 to 20. We repeat each experiment for 20 runs with different random seeds and calculate the average value. The confidence intervals with 95% confidence level are given to indicate the reliability of the simulation results. The main system parameters are listed in Table II. We consider both *single hop* and *multi-hop* network scenarios.

1) *Single-hop scenario*: we simulate a system with 30 users and 2 channels. All users are within the communication range of each other. we apply the hopping sequence $H = [c_1\ c_1\ c_2\ c_1\ c_2\ c_2\ c_2]$ where c_1 and c_2 represent two available channels. S-D pairs are randomly selected from the users in the system.
2) *Multi-hop scenario*: we simulate a system with 8 channels and 200 users which are randomly distributed in a 2 km $\times$ 2 km area. The transmission range of each user is 100m. Therefore, some users are outside the communication range of other ones, and thus it is possible for multiple S-D pairs to communicate over the same channel simultaneously without interfering with each other. S-D pairs are randomly selected within one-hop neighbors in the system. We use the 8-channel hopping sequence derived in the Section III.

A. Performance evaluation in the single-hop scenario: throughput and channel utilization

Figure 4 shows the system throughput of the proposed DSMMAC in the single-hop scenario. It can be seen that the system throughput increases with the duration of each data transmission (i.e., T). With a larger T, an S-D pair can occupy the channel for a longer time when they successfully negotiate, which improves the transmission efficiency due to less overhead per transmission in terms of time percentage, and thus obtains a larger throughput.

Figure 5 shows the channel utilization with different numbers of S-D pairs. Due to the nice properties of DSMMAS (e.g., parallel rendezvous and asynchronization), multiple S-D pairs attempting to access the media are efficiently distributed in the frequency domain. Meanwhile, the time instances that multiple source nodes send RTS message are separated in the time domain, which benefits for the improvement of channel utilization with the increase of S-D pairs. The channel utilization (i.e., the probability of starting a new data transmission over a channel) increases with the number of S-D pairs as well. The analytical results match well with the simulation ones, especially when the average transmission time T is large. This is because the inaccuracy introduced by converting the continuous exponential distribution to the discrete geometric distribution becomes negligible when T is large.

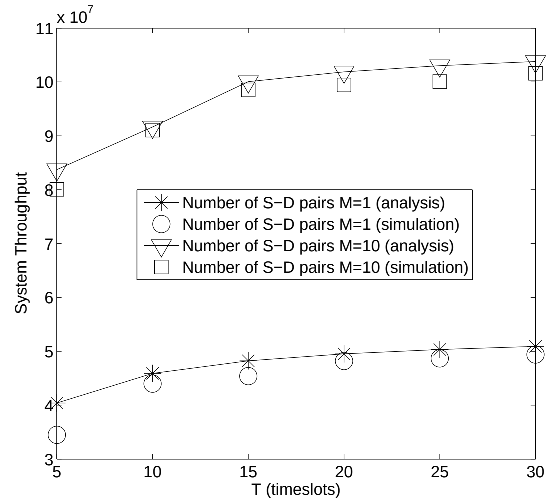


Fig. 4. System throughput versus duration of each data transmission.

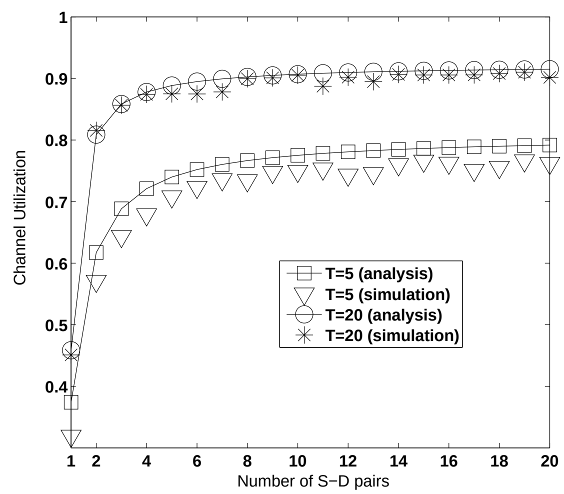


Fig. 5. Channel utilization with different number of S-D pairs.

B. Performance evaluation in the multi-hop scenario: throughput, access delay, and overhead

Figures 6 - 8 show the performance of the proposed DSMMAC in the multi-hop scenario. To further verify the efficiency of the proposed DSMMAC, we compare it with a multi-channel MAC based on the hopping sequence proposed by Dasilva and Guerreiro in [20] (Denoted as DG scheme). In DG scheme, a pre-defined hopping sequence is adopted by all users to reduce the overhead of exchanging hopping information among users and eliminate the synchronization

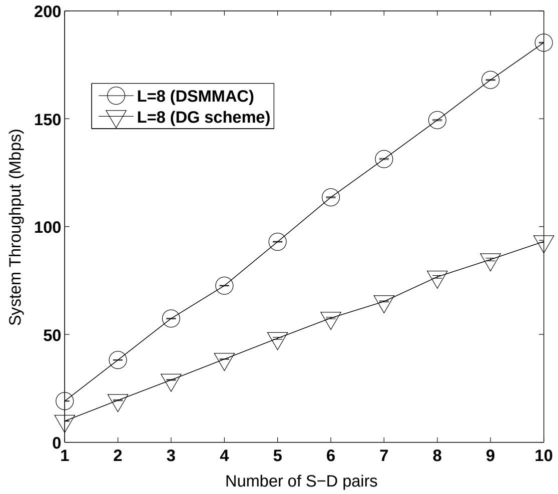


Fig. 6. System throughput with different number of S-D pairs.

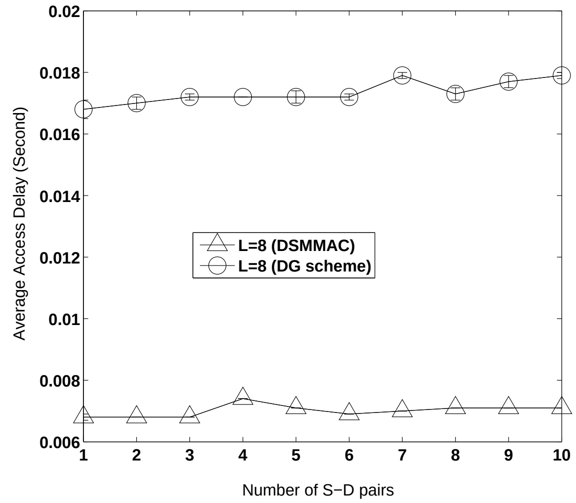


Fig. 7. Medium access delay with different number of S-D pairs.

requirement. The comparison between our DSMMAC and DG scheme is fair due to similar features and objectives of both schemes.

Figure 6 shows the system throughput with different number of S-D pairs. We observe that the total system throughput increases with the number of S-D pairs in both DSMMAC and DG scheme, due to the nice features of multiple rendezvous and asynchronous transmissions. In addition, the proposed DSMMAC outperforms DG scheme. The system throughput of DSMMAC is twice of that of DG scheme when the number of S-D pairs is 10.

Figure 7 shows the channel access delay, which is defined as the average duration from the time instant that a source node attempts to access the channel to the time it successfully communicates with its destination node. We observe that the channel access delay increase slightly with the increasing of S-D pairs. With a larger number of S-D pairs, the probability that an S-D pair fails to negotiate increases due to a busy channel increases, which leads to a longer access delay. In addition, the proposed DSMMAC outperforms the DG scheme in terms of access delay due to a higher rendezvous probability of our designed difference set based hopping sequence. It can be seen that DSMMAC achieves around 150% reduction in channel access delay compared with DG scheme.

Figure 8 shows the throughput on each channel (not each S-D pair). Recall that we have used the 8-channel hopping sequence defined at the end of Section III. It can be seen that channel c_1 achieves a higher throughput than other channels. This is because we allocate channel c_1 to the remaining time slot in the hopping sequence design, which means that channel c_1 will be used more often than the other channels. The achieved throughputs on channels c_2 to c_8 are similar, which demonstrate the fair use across channel resources.

For the proposed DSMMAC, each source node sends a RTS message over its hopping channel at the beginning of a time slot to probe whether its destination node hops on the same channel. This probing process may occur multiple times

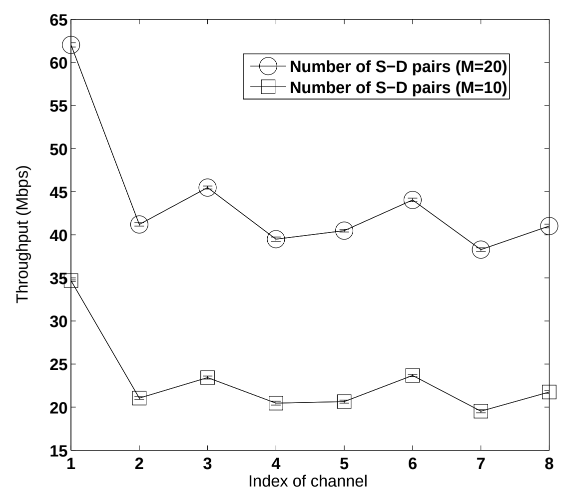


Fig. 8. Achieved throughput over each channel.

before an S-D pair successfully negotiates, which may lead to a significant overhead in terms of power consumption. Figure 9 shows this overhead, which is defined as the ratio of the power consumed by the probing process to the power consumed by the probing process and data transmission. It can be seen that with a longer average data transmission time T, the power consumption overhead decreases. On the other hand, a larger number of S-D pairs (i.e., M) leads to a longer probing time for each S-D pair, and thus increases the overhead. Compared with the DG scheme, the proposed DSMMAC significantly decreases the power consumption overhead under the same setting of system parameters.

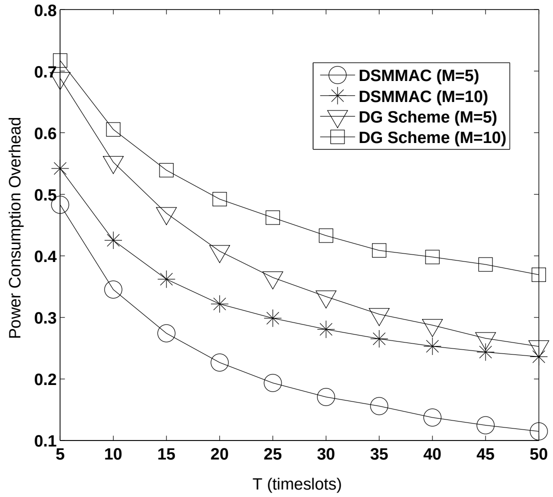


Fig. 9. Power consumption overhead versus the duration of each data transmission.

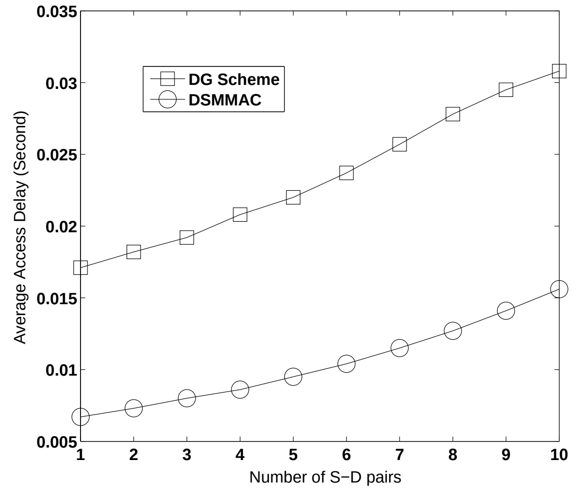


Fig. 11. Medium access delay with different number of S-D pairs.

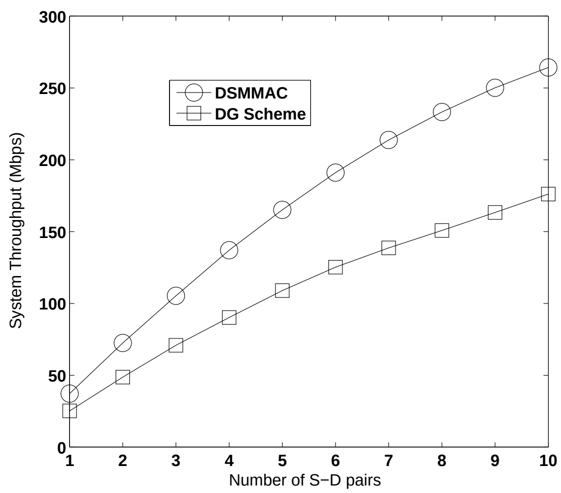


Fig. 10. System throughput with different number of S-D pairs.

C. Performance evaluation with deterministic data transmission time: throughput and access delay

The performance analysis and the above simulations assume exponential distribution of the data transmission time. Next we show the performance of the proposed scheme based on deterministic data transmission time (20 timeslots) in Figures 10 and 11. It can be seen the similar performance improvement of the proposed DSMMAC in terms of the system throughput and access delay.

VII. Related work

As the promising OFDM technology opens a door for multi-channel system, MAC protocol design for multi-channel systems has become a critical research issue and attracted great attention recently. Some previous works use a dedicated control channel for control message exchange to coordinate data transmissions over multiple data channels in the multi-channel MAC design [12] - [16]. [13] proposed a dynamic channel selection scheme, using one dedicated frequency channel for control message exchanges. [14] proposed a cooperative multi-channel MAC, using a control channel group to facilitate the channel selection. [15] proposed an adaptive admission control and channel selection scheme, using a secondary network controller to collect the information on the available channels and broadcast the allocation results through a local common control channel.

Another form of dedicated control channel was in the time domain, i.e., control and data messages were transmitted at different time slots. The multi-channel MAC in [16] divided the time into an alternating sequence of control and data phases. Since no data transmission was allowed in the idle data channels during the control phases, such protocol limited the achievable network throughput when the number of channels or the length of control phase was large.

Most multi-channel MAC protocols required either tight or loose synchronization among all nodes. In the channel hopping multiple access scheme in [17], all nodes used a predefined common hopping pattern, which eliminated heavy signaling overhead, but it still required tight synchronization among users. A MAC protocol with priority based channel access is proposed in [18] to reduce the node contention and then improve the channel utilization, in which the loose synchronization among neighboring nodes is needed in the implementation.

There are MAC protocols requiring the sources to know the hopping information of the destinations, which may involve significant signaling overheads. In the multiple rendezvous MAC protocol in [19], each node followed multiple hopping sequences in a time-multiplexed manner. When a node attempted to initiate a transmission to another node, it waited in a channel until their rendezvous arose over this channel. However, to make rendezvous happen, the sources needed to know their destinations' current hopping sequences via a seed broadcast mechanism, which increased the signaling overhead and degraded the system performance. Moreover, the reliability of broadcast transmissions could not be guaranteed over an error-prone wireless channel due to the lack of acknowledgment mechanisms.

Some recent work designed efficient hopping sequences without frequent hopping information exchanges among nodes. [20] designed the hopping sequences that can ensure two nodes to meet without knowing each other's hopping sequence. It showed that a proper sequence design reduced the time to rendezvous compared with a simple random rendezvous. [8] proposed a quorum-based channel hopping (QCH) framework for the control channel establishment. The authors studied two optimal QCH systems in synchronous systems: one minimized the time to rendezvous and the other guaranteed the even distribution of the rendezvous points in both time and channel domains. Different from [8] and [20], our work focuses on designing a distributed and asynchronous MAC to provide efficient and robust channel access performances among different users over multiple channels. In addition, we analyze the protocol performance in terms of network throughput, access delay, channel utilization, and power consumption overhead.

VIII. Conclusions and Future Work

In this paper, we have proposed a difference set based asynchronous MAC protocol for multi-channel wireless networks. By allowing all users to use the same hopping sequence derived from difference sets, multiple source-destination pairs can rendezvous over different channels simultaneously in a distributed manner so that the utilization of a multi-channel system can be greatly improved. It has been demonstrated that the proposed MAC can achieve high system throughput, low access delay, and good fairness among users under various network conditions. In addition, compared with previous proposed protocols, our protocol requires neither a dedicated control channel nor global synchronization among users, and thus it is promising for practical deployment.

There are several interesting issues from this work. For instance, in a cognitive radio network, the channel availability and channel conditions of secondary users may differ from each other as they are distributed over various geographical locations. How to design hopping sequences to adapt to the dynamics and heterogeneity of the available spectrum bands to the secondary users is very important.

Acknowledgment

This work is partially supported by a discovery grant from NSERC, Canada, and the General Research Funds (Project Number 412509) established under the University Grant Committee of the Hong Kong Special Administrative Region, China.

References

[1] D. Valerio, F. Ricciato, and P. Fuxjaeger, "On the feasibility of IEEE 802.11 multi-channel mutli-hop mesh networks", *Computer Communications*, vol. 31, pp. 1484-1496, 2008.

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

[3] R. Maheshwari, H. Gupta, and S. R. Das, "Multichannel MAC protocols for wireless networks", in *Proc. IEEE SECON*, USA, Sept. 2006.

[4] J. Mo, H. W. So, and J. Walrand, "Comparison of multi-channel MAC protocols", *IEEE Trans. Mobile Computing*, vol. 7, no. 1, pp. 50-65, Jan. 2008.

[5] Y. Bi, K. H. Liu, L. X. Cai, X. Shen, and H. Zhao, "A Multi-Channel Token Ring Protocol for QoS Provisioning in Inter-Vehicle Communications" *IEEE Transactions on Wireless Communications*, vol. 8, no. 11, pp. 5621-5631, Nov. 2009.

[6] J.-R. Jiang, Y.-C. Tseng, C.-S. Hsu, and T.-H. Lai, "Quorum-based asynchronous power-saving protocols for IEEE 802.11 ad hoc networks", *ACM Journal on Mobile Networks and Applications*, vol. 10, nos. 1-2, pp. 169-181, 2005.

[7] R. Zheng, J. C. Hou, and L. Sha, "Optimal block design for asynchronous wake-up schedulers and its applications in multihop wireless networks", *IEEE Trans. on Mobile Computing*, vol. 5, no. 9, pp. 1228-1241, Sept. 2006.

[8] K. Bian, J.-M. Park, and R. Chen, "A Quorum-based framework for establishing control channels in dynamic spectrum access networks," in *Proc. MobiCom*, China, 2009.

[9] P. Dutta and D. Culler, "Practical asynchronous neighbor discovery and rendezvous for mobile sensing applications", in *Proc. ACM Conf. Embedded Network Sensor Systems*, USA, Nov. 2008.

[10] L.D. Baument, Lecture Notes in Mathematics, Cyclic Difference Sets, Springer, 1971.

[11] Ian Aderson, Combinatorial Designs and Tournaments, Oxford University Press, 1997.

[12] T. Luo, M. Motani, V. Srinivasan, "Cooperative asynchronous multi-channel MAC: design, analysis, and implementation", *IEEE Trans. on Mobile Computing*, vol. 8, no. 3, pp. 338-352, March 2009.

[13] J. Li, D. Zhang, S. Ji, and L. Guo, "RCS: a random channel selection with probabilistic backoff for multi-channel MAC protocols in WSNs," in *Proc. IEEE GLOBECOM*, USA, Dec. 2010.

[14] Y. Moon and V. Syrotiuk, "A cooperative dual access multi-channel MAC protocol for ad hoc networks," in *Proc. IEEE GLOBECOM*, USA, Dec. 2010.

[15] A. Alshamrani, L. Xie, and X. Shen, "Adaptive admission-control and channel-allocation policy in cooperative ad hoc opportunistic spectrum networks," *IEEE Trans. on Vehicular Technology*, vol. 59, no. 5, pp. 1618-1629, May 2010.

[16] J. So and N. Vaidya, "Multi-channel MAC for ad hoc networks: handling multi-channel hidden terminal using a single transceiver", in *Proc. ACM MobiHoc*, Japan, May 2004.

[17] A. Tzamaloukas and J. J. Carcia-Luna-Aceves, "Channel-hopping multiple access", in *Proc. IEEE ICC*, USA, June 2000.

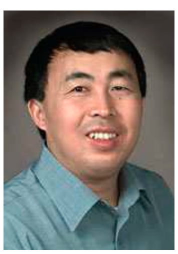
[18] J. Zhu, Z. Wang, and D. Xu, "A unified MAC and routing framework for multichannel multi-interface ad hoc networks," *IEEE Trans. on Vehicular Technology*, vol. 59, no. 9, pp. 4589-4601, Nov. 2010.

[19] P. Bahl, R. Chandra, and J. Dunagan, "SSCH: slotted seeded channel hopping for capacity improvement in IEEE 802.11 ad hoc wireless networks", in *Proc. ACM MobiCom*, USA, Sept. 2004.

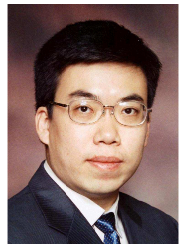
[20] L.A. Dasilva, and I. Guerreiro, "Sequence-based rendezvous for dynamic spectrum access", in *Proc. IEEE Dynamic Spectrum Access Networks (DySPAN)*, USA, Oct. 2008.

Fen Hou received the Ph.D. degree in electrical and computer engineering from the university of Waterloo, Waterloo, Canada, in 2008. She is currently a postdoctoral fellow in the Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong, China. Her research interests include resource allocation, connection admission control, scheduling scheme, and QoS provisioning for multimedia communications.

Lin X. Cai received the M.A.Sc. and Ph.D. degrees in Electrical and Computer Engineering from the University of Waterloo, Ontario, Canada, in 2005 and 2009, respectively. In 2010, She was awarded a Postdoctoral Fellowship from the Natural Sciences and Engineering Research Council of Canada (NSERC) and worked as postdoctoral fellow in the Department of Electrical and Computer Engineering, McMaster University, Ontario, Canada. Since 2011, she has been a postdoctoral research fellow in the Department of Electrical Engineering, Princeton University, New Jersey, USA. Her current research interests include resource management for broadband multimedia networks, green communication and networking, and cross-layer optimization and QoS provisioning.



Xuemin (Sherman) Shen received the B.Sc.(1982) degree from Dalian Maritime University (China) and the M.Sc. (1987) and Ph.D. degrees (1990) from Rutgers University, New Jersey (USA), all in electrical engineering. He is a Professor and University Research Chair, Department of Electrical and Computer Engineering, University of Waterloo, Canada. Dr. Shen's research focuses on resource management in interconnected wireless/wired networks, UWB wireless communications networks, wireless network security, wireless body area networks and vehicular ad hoc and sensor networks. He is a co-author of three books, and has published more than 400 papers and book chapters in wireless communications and networks, control and filtering. Dr. Shen served as the Technical Program Committee Chair for IEEE VTC'10, the Symposia Chair for IEEE ICC'10, the Tutorial Chair for IEEE ICC'08, the Technical Program Committee Chair for IEEE Globecom'07, the General Co-Chair for Chinacom'07 and QShine'06, the Founding Chair for IEEE Communications Society Technical Committee on P2P Communications and Networking. He also served as a Founding Area Editor for *IEEE Transactions on Wireless Communications*; Editor-in-Chief for *Peer-to-Peer Networking and Application*; Associate Editor for *IEEE Transactions on Vehicular Technology*; *Computer Networks*; and *ACM/Wireless Networks*, etc., and the Guest Editor for *IEEE JSAC*, *IEEE Wireless Communications*, *IEEE Communications Magazine*, and *ACM Mobile Networks and Applications*, etc. Dr. Shen received the Excellent Graduate Supervision Award in 2006, and the Outstanding Performance Award in 2004 and 2008 from the University of Waterloo, the Premier's Research Excellence Award (PREA) in 2003 from the Province of Ontario, Canada, and the Distinguished Performance Award in 2002 and 2007 from the Faculty of Engineering, University of Waterloo. Dr. Shen is a registered Professional Engineer of Ontario, Canada, an IEEE Fellow, an Engineering Institute of Canada fellow, and a Distinguished Lecturer of IEEE Communications Society.



Jianwei Huang has been an Assistant Professor in the Department of Information Engineering at the Chinese University of Hong Kong since 2007. He received B.S. in Electrical Engineering from Southeast University (Nanjing, Jiangsu, China) in 2000, M.S. and Ph.D. in Electrical and Computer Engineering from Northwestern University (Evanston, IL, USA) in 2003 and 2005, respectively. He worked as a Postdoc Research Associate in the Department of Electrical Engineering at Princeton University during 2005-2007, and as a summer intern in Motorola (Arlington Heights, IL, USA) in 2004 and 2005. Dr. Huang leads the Network Communications and Economics Lab (ncel.ie.cuhk.edu.hk) at the Chinese University of Hong Kong. He conducts research in the area of nonlinear optimization and game theoretical analysis of communication networks, with current focus on network economics, cognitive radio networks, broadband communication networks, and multimedia over wireless. He is the recipient of the IEEE GLOBECOM Best Paper Award in 2010, the IEEE ComSoc Asia-Pacific Outstanding Young Researcher Award in 2009, Asia-Pacific Conference on Communications Best Paper Award in 2009, and Walter P. Murphy Fellowship at Northwestern University in 2001.

Dr. Huang is or has served as Editor of *IEEE Transactions on Wireless Communications* (2010-), Associate Editor of (Elsevier) *Journal of Computer and Electrical Engineering* (2007-2010), the Lead Guest Editor of *IEEE Journal of Selected Areas in Communications* special issue on "Game Theory in Communication Systems", a Guest Editor of (Wiley) *Wireless Communications and Mobile Computing* special issue on "Innovative Communications for a Better Future", the Lead Guest Editor of *Journal of Communications* special issue on "Cognitive-Radio Enabled Communications and Networking", the Lead Guest Editor of *Journal of Advances in Multimedia* special issue on "Collaboration and Optimization in Multimedia Communications", and a Guest Editor of *Journal of Advances in Multimedia* special issue on "Cross-layer Optimized Wireless Multimedia Communications".

Dr, Huang is or has served as Vice Chair of IEEE Communications Society Multimedia Communications Technical Committee (2010-2012), Director of IEEE Communications Society Multimedia Communications Technical Committee E-letter (2010), the Student Activities Co-Chair of International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt) 2011, the TPC co-chair of the International Conference on Game Theory for Networks (GameNets) 2009, the TPC co-chair of IEEE Globecom Wireless Communications Symposium 2010, the TPC co-chair of the International Wireless Communications and Mobile Computing Conference (IWMobile Computing Symposium 2010, and TPC member of many conferences such as INFOCOM, MobiHoc, ICC, GLOBECOM, DySPAN, WiOpt, and WCNC.