Channel-Hopping Scheme and Channel-Diverse Routing in Static Multi-Radio Multi-Hop Wireless Networks

Ting-Yu Lin, *Member*, *IEEE*, Kun-Ru Wu, *Student Member*, *IEEE*, and Guang-Chuen Yin

Abstract—In modern wireless networks with multiple orthogonal (non-overlapping) channels available, one essential performance topic is how to effectively exploit channel diversity to enable parallel communications. Generally, having a radio interface hop through all available channels produces better spectrum diversity than binding it permanently to one channel, at the cost of channel switching delays and potentially compromised network connectivity. Moreover, multi-hop communications become challenging due to the lack of a common rendezvous for discovering routes and the difficulty of relaying packets from hop to hop. In this paper, we propose a multi-radio channel-hopping scheme (CHS) that preserves network connectivity. We prove that less than three radios are required by CHS in order to achieve good channel overlapping in widespread IEEE 802.11-based wireless systems. Corresponding channel-diverse routing (CDR) protocol is devised to realize efficient multi-hop communications. Simulation results demonstrate that the proposed CDR outperforms other strategies in static IEEE 802.11a multi-hop networking environments.

Index Terms—Channel diversity, multi-channel routing, multi-radio multi-hop wireless network, IEEE 802.11, prime field

1 Background

With the advent of ever-growing bandwidth-demanding services and applications, throughput performance remains a critical issue in the wireless networking community. One promising way for throughput improvement is to utilize multiple orthogonal (non-overlapping) channels available in a wireless system. Take the dominant wireless standard IEEE 802.11 for instance, three and eight (up to twelve) orthogonal channels can be utilized in IEEE 802.11b/g and 802.11a respectively. However, design of multi-channel protocols (in both single-radio and multi-radio networks) is nontrivial and deals with not only the link-layer channel assignment scheme but also the routing strategy for multi-hop communications [14]. Below we review several past multi-channel research works in the literature.

A static channel allocation scheme is proposed to bind each radio interface to a channel permanently (or for a long term) in [6]. The static binding is straightforward and indeed utilizes multiple channels simultaneously, but this approach requires as many radio interfaces as available channels, which is barely possible in real practices. As a result, several optimization models, targeting on wireless mesh networks (WMNs), are designed to enhance network performance under the constraint of limited radio interfaces [2], [17], [22]. These works also permanently (statically) bind each radio to a dedicated channel. Since the majority of data packets in a WMN are destined for or originated from the Internet via gateways, the multi-channel protocols can thus be devised according to this unique traffic pattern. However, in general multi-hop wireless networks, communications can be any source-destination pair and static binding lacks the flexibility of reaching all available channels especially when traffic patterns are dynamic and unpredictable.

Rather than statically binding each interface, some multi-channel protocols are proposed to dedicate one radio interface to a *control channel* (CC) and have the remaining one or more radios switch between usable *data channels* (DCs) [9], [12], [26]. Whenever there is a communications need, the intended transmitter and receiver negotiate a DC to use via CC in an on-demand manner. Two attendant drawbacks come with this approach. First, depending on the number of DCs, the CC may become underutilized or too congested (bottleneck effect). Second, communications overhead is high, since the negotiation is exercised on a per-packet basis. To overcome the control channel (CC) problems, some propose to have one interface fixed/assigned to a (data) channel permanently (or relatively a long time) for receiving data, and the remaining interface(s) switch between other channels [15], [20]. The fixed interface is called *receiving interface*, and the remaining as *switchable interface(s)*. Without a CC, an intended transmitter tunes its switchable interface to the channel used by the receiving interface of an intended receiver. In this way, the CC underutilization and bottleneck problems can be avoided, but the performance now greatly depends on the receiving channel allocation scheme. If a receiving interface is not configured wisely, the system performance can be negatively compromised.

Next, we introduce another genre of multi-channel protocols that requires time synchronization between nodes.

• *The authors are with the Department of Electrical and Computer Engineering, National Chiao Tung University, Hsinchu 300, Taiwan. E-mail: ting@cm.nctu.edu.tw, wufish.cm96g@nctu.edu.tw, gcyin831@gmail.com.*

Manuscript received 29 Dec. 2012; revised 06 Aug. 2013; accepted 22 Sep. 2013. Date of publication 07 Oct. 2013; date of current version 12 Dec. 2014. (Corresponding author: T.-Y. Lin.)
Recommended for acceptance by Y. Yang.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TC.2013.199

In [23], the authors propose a multi-channel protocol using a single transceiver (radio interface). The idea is to embed a negotiation phase in the ATIM (Ad-hoc Traffic Indication Map) window that is periodically sent under the Power Save Mode (PSM). Every node has to hop on a predefined control channel (CC) when entering the ATIM window. The negotiation phase is to determine a data channel (DC) to use after the ATIM window finishes. This approach, termed as *split phase* based on the classifications in [19], incurs both the CC problems and high overhead as most CC-based protocols do. Without using a CC, channel-hopping schemes are another alternative for single-radio multi-channel protocols [24], [25]. Dividing the time axis into virtual slots, all nodes perform channel switching by following a *common hopping sequence*. Two nodes stop hopping and start transmitting on the current channel whenever there is a communications need. This common hopping approach obviously guarantees network connectivity, but becomes inefficient when multiple nodes attempt to transmit simultaneously in the neighborhood. As a result, authors in [3] propose to use *distinct hopping sequences*, governed by randomly chosen (channel, seed) pairs. By using a prime number of channels and having nodes cycle through all channels based on predefined hopping offsets (seeds), this approach nicely preserves the network connectivity while enabling simultaneous transmissions in the neighborhood. However, due to lack of a CC, broadcast transmissions become difficult and consequently multi-hop routing packets cannot efficiently propagate. Moreover, channel switching overhead should also be considered in channel-hopping schemes.

In this paper, we adopt the channel-hopping scheme with distinct hopping sequences to better exploit channel diversity. Several major contributions are made in this work. First, *we propose a channel-hopping scheme (CHS) for multi-radio networking environments*. CHS leverages the connectivity-preserving methodology introduced in [3] and extends the method to multi-radio hopping sequences design. Our goal is to increase the overlapping ratio between nodes by utilizing multiple radios while maintaining reasonable channel randomness. Intuitively more radios lead to more overlapping time slots. Given p orthogonal channels used by CHS, we prove that $\sqrt{p}$ radios (tantamount to less than three radios in IEEE 802.11-based systems) are required to achieve the best case of overlapping. Second, *corresponding channel-diverse routing (CDR) is devised to realize efficient multi-hop data delivery under this channel-hopping system*. We propose to embed a broadcast slot running over a control channel (CC) to facilitate broadcast transmissions. The CC is dedicated to exchanging control information, which prevents CDR from having the congestion (bottleneck) problem due to frequent data channel negotiations. In fact, the usage of CC is beneficial in providing a common rendezvous for control packets, such as ad-hoc beacons, hopping table exchange messages, routing requests, etc. For route selections, CDR takes both channel quality and overlapping efficiency from hop to hop into consideration, and decides on a route with the minimum average expected transmission time (ETT) from source to destination. Third, *we propose to implement per-destination queues and perform selective round-robin (SRR) mechanism for queue services*. In channel-hopping schemes, data queue management is critical in avoiding blocking other transmittable packets due to a HOL (Head-of-Line) packet whose receiver is currently out of reach. By exercising proper data queue management, we aim to transmit as many packets as possible in a single time slot.

The remainder of this paper is organized as follows. In Section 2, we state the assumptions made by this work and the key problems that pertain to our designs. Sections 3 and 4 detail the proposed channel-hopping scheme (CHS) and channel-diverse routing (CDR) protocol, respectively. Simulation results are presented in Section 5 for performance evaluation and comparison with other multi-channel approaches. Finally, Section 6 draws our concluding remarks.

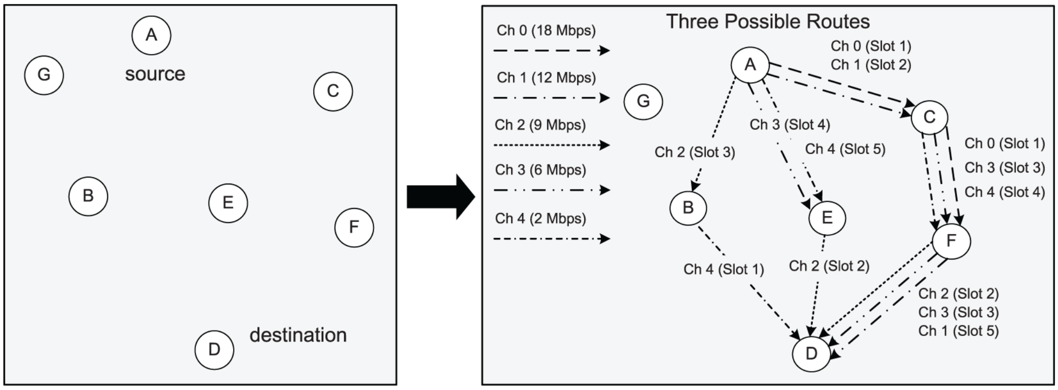
2 Assumptions and Problem Statement

Before elaborating on the proposed multi-channel protocols, we make the following assumptions for the target wireless environment.

- All nodes use half-duplex radio interfaces (transceivers) that are switchable between channels. When switching to a different channel, a transceiver takes about 80 to 200 200 s, depending on the hardware technology, for its circuits to stabilize [1], [10]. Adding to this delay is the link-layer waiting time, which is necessary for mitigating the multi-channel hidden-terminal problem, collectively we define the time spent on one channel switching the *switching delay*, denoted as t_{SDE}.
- We divide the time axis into time slots with equal *slot duration*, denoted as t_{SDU}. All nodes implement time synchronization mechanism based on reference broadcasts in order to run the channel-hopping scheme [8].
- Given p data channels, denoted as $C_0, C_1, \ldots, C_{p-1}$, the achievable rate R_i associated with channel C_i, where $0 \leq i < p$, can be estimated through packet probing mechanism introduced in [5][1].

Fig. 1 illustrates a sample single-radio network with five orthogonal channels (Ch 0, Ch 1, ..., Ch 4) having different link rates (ranging from 2 to 18 Mbps). The time axis is divided into five-slot cycles in order to go through the five channels. Each node randomly chooses its own channel-hopping schedule as displayed in Fig. 1 (lower left). Unfortunately, because of disjoint channel-hopping schedules chosen, nodes A and G become disconnected despite the fact that they are within the communications range, which is not desirable in a general-purpose wireless network. Thus the first problem we face is *how to design a channel-hopping scheme that allows nodes to independently decide their own channel schedule and preserves network connectivity at the same time?* Now, suppose node A (source) needs to communicate with node D (destination) and three possible routes are discovered: $\mathrm{A} \rightarrow \mathrm{B} \rightarrow \mathrm{D}$, $\mathrm{A} \rightarrow \mathrm{E} \rightarrow \mathrm{D}$, and $\mathrm{A} \rightarrow \mathrm{C} \rightarrow \mathrm{F} \rightarrow \mathrm{D}$. As shown in Fig. 1 (right), for the first hop of route $\mathrm{A} \rightarrow \mathrm{B} \rightarrow \mathrm{D}$, nodes A and B can only communicate at Slot 3 over middle-quality Ch 2. On the other hand, nodes A and E in route $\mathrm{A} \rightarrow \mathrm{E} \rightarrow \mathrm{D}$ have two overlapping slots (Slot 4 and 5) over low-quality Ch 3 and Ch 4, while nodes A and C in route $\mathrm{A} \rightarrow \mathrm{C} \rightarrow \mathrm{F} \rightarrow \mathrm{D}$ also have two

1. Note that R_i indicates the channel capacity (maximal achievable link rate) for C_i. Our packet probing mechanism constantly measures and updates data rates for adjacent links over channel C_i according to current communication distance, background noise, and channel conditions between the intended transmitter and receiver.



Random Channel Schedules

Node \ Slot	SDU 1	SDE	SDU 2	SDE	SDU 3	SDE	SDU 4	SDE	SDU 5	SDE
A	Ch 0		Ch 1		Ch 2		Ch 3		Ch 4	
B	Ch 4		Ch 3		Ch 2		Ch 1		Ch 0	
C	Ch 0		Ch 1		Ch 3		Ch 4		Ch 2	
D	Ch 4		Ch 2		Ch 3		Ch 0		Ch 1	
E	Ch 1		Ch 2		Ch 0		Ch 3		Ch 4	
F	Ch 0		Ch 2		Ch 3		Ch 4		Ch 1	
G	Ch 1		Ch 2		Ch 3		Ch 4		Ch 0	

A and G are disconnected due to disjoint channel schedules

Route A → B → D

Node \ Slot	SDU 1	SDE	SDU 2	SDE	SDU 3	SDE	SDU 4	SDE	SDU 5	SDE
A	Ch 0		Ch 1		Ch 2		Ch 3		Ch 4	
B	Ch 4		Ch 3		Ch 2		Ch 1		Ch 0	
D	Ch 4		Ch 2		Ch 3		Ch 0		Ch 1	

Route A → E → D

Node \ Slot	SDU 1	SDE	SDU 2	SDE	SDU 3	SDE	SDU 4	SDE	SDU 5	SDE
A	Ch 0		Ch 1		Ch 2		Ch 3		Ch 4	
E	Ch 1		Ch 2		Ch 0		Ch 3		Ch 4	
D	Ch 4		Ch 2		Ch 3		Ch 0		Ch 1	

Route A → C → F → D

Node \ Slot	SDU 1	SDE	SDU 2	SDE	SDU 3	SDE	SDU 4	SDE	SDU 5	SDE
A	Ch 0		Ch 1		Ch 2		Ch 3		Ch 4	
C	Ch 0		Ch 1		Ch 3		Ch 4		Ch 2	
F	Ch 0		Ch 2		Ch 3		Ch 4		Ch 1	
D	Ch 4		Ch 2		Ch 3		Ch 0		Ch 1	

Fig. 1. Illustration of the multi-channel issues in a multi-hop wireless network, where each node has single radio with randomly-chosen channel-hopping schedule repeated in five-slot cycles and five orthogonal channels are available in the system.

overlapping slots (Slot 1 and 2) over high-quality Ch 0 and Ch 1. More overlapping slots imply more delivery opportunities. Meanwhile, with the same number of overlapping slots, channels having better qualities (higher link rates) are preferred because more packets may be delivered in a single time slot. Consequently, in our sample network, route $A \rightarrow C \rightarrow F \rightarrow D$ can potentially act as the best route among the three, despite that it has one more hop than the other two routes. The notion that hop count no longer serves as the single metric in rate-diverse multi-hop networks has been investigated in [7]. Therefore the second problem we need to solve is *how to devise a multi-hop routing protocol that factors in both the overlapping conditions and channel qualities (link rates) to determine a best route under such channel-hopping environment?*

Based on the foregoing observations, we aim to develop a connectivity-preserving channel-hopping scheme (CHS) for the first problem, and an efficient channel-diverse routing (CDR) for the second problem. Note that in practical wireless networks, channel allocation and routing mechanisms tend to interact with each other. As a result, we propose to design a complete multi-channel protocol suite that takes care of both mechanisms in a unified framework.

3 Channel-Hopping Scheme (CHS)

In this section, we propose a channel-hopping scheme (CHS) that guarantees network connectivity. The channel overlapping behavior in CHS follows the properties of *prime field*, subfield of the Galois field, which is reviewed in Section 3.1. Based on the arithmetic operations of some prime field, Section 3.2 and 3.3 present how to configure channel-hopping schedule for each interface in single-radio and multi-radio networks respectively[2]. Mathematical analysis is provided in Section 3.4 to investigate the overlapping properties of our channel-hopping design and provide insight into the relationship of available number of channels and required radio interfaces.

3.1 Properties of the Galois Field

Fields are abstractions of number systems (set $\mathbb{F}$) and their essential operations (addition and multiplication) [11]. Familiar examples of fields include the rational numbers, the real numbers, and the complex numbers, all with infinite elements in $\mathbb{F}$. If the set $\mathbb{F}$ is finite, then the field is said to be *finite*. For the finite fields, Galois proved their existence and concluded that *the number of elements in a finite field must be a prime power, say $q = p^r$, where p is a prime and r is a positive integer*. Define the Galois field $\mathbb{GF}(q)$. If $r = 1$, then the Galois field becomes $\mathbb{GF}(p)$, known as the *prime field*. The elements $0, 1, 2, \ldots, p-1$ then form a set $\mathbb{F}$ in prime field $\mathbb{GF}(p)$, which is isomorphic to *the integers modulo p* (where p is called the modulus).

In a prime field $\mathbb{GF}(p)$, the addition operation $\oplus$ and multiplication operation $\odot$ are defined as follows.

Definition 1. *$\forall\ a, b \in \mathbb{F}$, $a \oplus b \triangleq (a+b)$ mod p and $a \odot b \triangleq (a \cdot b)$ mod p, where for any integer x, x mod p*

2. In the literature, the usage of quorum-based concept is another possible direction for designing overlapped channel-hopping schedules in single-radio wireless systems [4], [16], [18]. In this paper, we adopt the prime-based hopping-sequence design for our CHS and extend it to support multi-radio wireless systems.

Additive Inverse (Negative)

- 0 = 0
- 1 = 4
- 2 = 3
- 3 = 2
- 4 = 1

⊕	0	1	2	3	4
0	0	1	2	3	4
1	1	2	3	4	0
2	2	3	4	0	1
3	3	4	0	1	2
4	4	0	1	2	3

Addition in GF(5)

⊙	0	1	2	3	4
0	0	0	0	0	0
1	0	1	2	3	4
2	0	2	4	1	3
3	0	3	1	4	2
4	0	4	3	2	1

Multiplication in GF(5)

Multiplicative Inverse

$1^{-1} = 1$
$2^{-1} = 3$
$3^{-1} = 2$
$4^{-1} = 4$

Fig. 2. Operations of addition and multiplication in $\mathbb{GF}(5)$.

denotes the unique integer remainder r obtained upon dividing x by p $(0 \leq r \leq p-1)$, known as the reduction modulo p.

Like the usual integer operations, addition and multiplication in a prime field follow the associative, commutative, distributive laws, and the operation results are also elements in $\mathbb{F}$ (closeness property). Additive identity is 0, whereas multiplicative identity is 1. Subtraction of field elements is defined in terms of addition, while division of filed elements is defined in terms of multiplication. Below we have the definition for subtraction and division operations in a prime field.

Definition 2. *$\forall\ a, b \in \mathbb{F},\ a - b = a \oplus (-b)$ where $-b$ is the additive inverse (negative) of b, and for $b \neq 0$, $a/b = a \odot b^{-1}$ where b^{-1} is the multiplicative inverse of b.Recall that the additive inverse $-b$ is an element in $\mathbb{F}$ such that $b \oplus (-b) = 0$ (0 is the additive identity), while the multiplicative inverse b^{-1} is an element in in $\mathbb{F}$ such that $b \odot b^{-1} = 1$ (1 is the multiplicative identity).*

Fig. 2 shows an example of prime field $\mathbb{GF}(5)$ with the addition $\oplus$ and multiplication $\odot$ operations, where the elements of $\mathbb{F}$ are $\{0, 1, 2, 3, 4\}$. Those shaded areas display the results that lead to the additive or multiplicative identity, such that both the additive and multiplicative inverse for each element can be easily obtained from a lookup in this table. Note that no multiplicative inverse for element 0 exists as defined in Definition 2. We will revisit these arithmetic operations of a prime field in Section 3.4.

3.2 Single-Radio Channel-Hopping Design

Suppose p channels are available, denoted as $C_0, C_1, \ldots, C_{p-1}$, and define $\chi^A[k]$ as the channel ID used at the k^{th} time slot in a cycle for node A, where $1 \leq k \leq p$. In this section, we consider the channel-hopping design for the single radio interface equipped at each node. Based on the elements $\{0, 1, \ldots, p-1\}$ contained in $\mathbb{F}$ of a prime field, theoretically $p!$ random sequences can be generated. However, two randomly generated sequences cannot guarantee overlapping among elements (as shown in Fig. 1). Thus we only generate *pseudo-random sequences*, for each of which a starting channel (SC) and hopping offset (Seed) are predefined, as the legitimate channel-hopping schedules. Specifically, if node A selects its starting channel (SC) as χ^A and hopping offset (Seed) as α^A, then all channels used in a cycle in order can be inferred by computing $\chi^A[k] = \chi^A \oplus \alpha^A(k-1)$, where $1 \leq k \leq p$. Three nice properties are possessed by pseudo-random sequences. First, since p is a prime, all channels are guaranteed to be visited *exactly once* in a cycle and repeated in the following cycles (i.e., $\chi^A[k + np] = \chi^A[k]$ for positive integer n), known as the periodicity property [11]. Second, the pseudo-random sequence allows a representation using a fixed-length (SC, Seed) pair, independent of number of available channels p. In other words, no matter how long a cycle can be, a (SC, Seed) pair is sufficient for expressing the channel-hopping schedule in use, which is beneficial in bounding the communications overhead spent on schedule exchange between nodes. Third, for any two nodes with different seed values, *one overlapping channel is guaranteed to occur in a cycle* (in-depth mathematical proof will be provided in Section 3.4). For instance, given $p = 5$, sequences $\{0, 1, 2, 3, 4\}$ with (SC, Seed) = (0, 1) and $\{3, 0, 2, 4, 1\}$ with (SC, Seed) $= (3, 2)$ are both legitimate channel-hopping schedules, which satisfy the periodicity property and overlap at the third slot on channel C_2.

Under this channel-hopping design, however, two nodes, say A and B, can still become disconnected if they happen to choose different starting channels $(\chi^A \neq \chi^B)$ but the same seed $(\alpha^A = \alpha^B)$. In this case, the two nodes always hop on distinct channels (separated by a distance of $|\chi^A - \chi^B|$) and never come into contact. To address this problem, a *Parity Slot* is introduced as Slot 0 operating on the parity channel β, which is set equal to the seed α. As a result, one more slot is added into a cycle, leading to $(p+1)$ slots, and the channel $\chi^A[k]$ visited at the k^{th} slot for node A can be derived as follows,

$$\chi^A[k] = \left\{ \begin{array}{ll} \alpha^A & \text{for } h = 0 \\ \chi^A \oplus \alpha^A \odot (h-1) & \text{for } 1 \leq h \leq p \end{array} \right\}, \qquad (1)$$

where $h = k \bmod (p+1)$.

Fig. 3 gives an example. Nodes A, B, and C choose their channel-hopping schedules by defining their own (SC, Seed) pairs. Unfortunately nodes B and C turn out to have disjoint hopping schedules because they choose different SCs with the same Seed. By inserting a Parity Slot, operating on the parity channel β, at the beginning of each cycle, nodes B and C are then made connected through Ch 2.

3.3 Multi-Radio Channel-Hopping Design

Now we extend the channel-hopping design to support multi-radio environments. Suppose w radio interfaces are available at each node. For some node A, denote these interfaces as $I_1^A, I_2^A, \ldots, I_w^A$. Our goal is to determine appropriate (SC, Seed) pair and the parity channel for each interface. Adopting similar notation system from Section 3.2, define χ_i^A, α_i^A, and β_i^A to represent the starting channel (SC), hopping offset (Seed), and parity channel for interface I_i^A at node A, where $1 \leq i \leq w$. First of all, we use the single-radio channel-hopping design for configuring the first interface I_1^A by setting χ_1^A, α_1^A to generate a pseudo-random sequence and letting

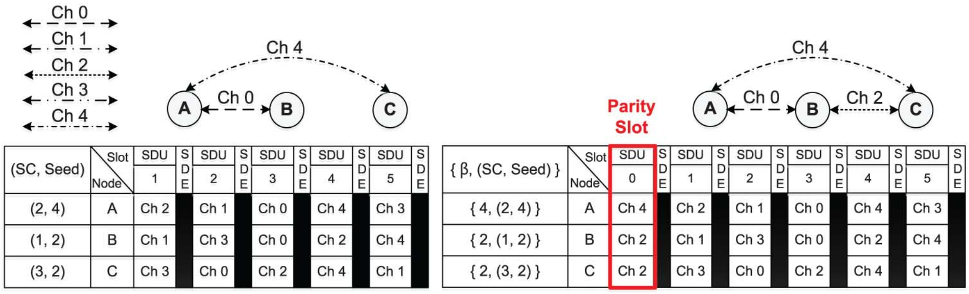


Fig. 3. Single-radio channel-hopping schedules for nodes A, B and C, where nodes B and C are disconnected due to disjoint hopping schedules (left). By introducing a Parity Slot (Slot 0), connectivity between nodes B and C can be preserved (right).

$\beta_1^A = \alpha_1^A$ for ensuring connectivity between disjoint schedules. Recall that two schedules are disjoint if they have different SCs with the same Seed. We then leverage this property for configuring the remaining of interfaces at node A, because we certainly do not want interfaces equipped at the same node to hop on the same channel at the same time slot. Specifically, for $2 \le i \le w$, α_i^A is made equal to α_1^A, and χ_i^A is suggested to be set as $\chi_i^A = \chi_1^A \oplus \alpha_i^A \odot (\lceil p/w \rceil \odot (i-1))$ with the objective of distributing the starting channels of all interfaces. Next we consider the parity channels β_i^A for $2 \le i \le w$. Since using the same channel at all interfaces in the parity slot is not desirable, we propose to set $\beta_i^A = \alpha_1^A \odot i$ to guarantee distinct channels are associated with all interfaces in the parity slot. This property is easy to prove (based on the fact that set $\{\beta_1^A, \beta_2^A, \ldots, \beta_w^A\}$ is a subset of $\{0, 1, 2, \ldots, p-1\}$ and each element in set $\{\beta_1^A, \beta_2^A, \ldots, \beta_w^A\}$ appears exactly once), considering $w \le p$ in practical multi-radio networks. Consequently, all radio interfaces have been configured with appropriate channel-hopping schedules, and the channel $\chi_i^A[k]$ visited by interface I_i^A at the k^{th} slot for node A can be obtained by the following,

$$\chi_i^A[k] = \begin{Bmatrix} \alpha_1^A \odot i & \text{for } h = 0 \\ \chi_1^A \oplus \alpha_1^A \odot (\lceil p/w \rceil \odot (i-1) \oplus (h-1)) & \text{for } 1 \le h \le p \end{Bmatrix}, \tag{2}$$

where $h = k \bmod (p+1)$.

Fig. 4 illustrates the channel schedules and overlapping conditions for two flows (A $\rightarrow$ B and C $\rightarrow$ D). Fig. 4(a) shows that, given $p = 5$ and $p = 11$, only one overlapping slot in a cycle is available for each flow when a single radio interface ($w = 1$) is equipped at each node. By adding one more radio interface, making $w = 2$, and exercising our CHS for configuring the two radios at each node, the overlapping slots (and usable channels) have been significantly increased. As displayed in Fig. 4(b), the high overlapping ratio is beneficial especially when the cycle is long (due to a large p). From the figure, we observe that CHS effectively enhances transmission opportunities between nodes at the cost of limited hardware (one more radio), while maintaining high level of channel diversity.

Algorithm 1 Channel Schedule Configuration at Node A

Given p channels and ω radio interfaces ($\omega \le p$);
Randomly select the starting channel χ_1^A from $[0, p-1]$;
Randomly select the seed α_1^A from $[0, p-1]$;
Set parity channel $\beta_1^A = \alpha_1^A$;
if ($\omega > 1$) **then**
  for ($i = 2$; $i \le \omega$; $i{+}{+}$) **do**
    Set $\alpha_i^A = \alpha_1^A$;
    Set $\chi_i^A = \chi_1^A \oplus \alpha_i^A \odot (\lceil \frac{p}{\omega} \rceil \odot (i-1))$;
    Set $\beta_i^A = \alpha_1^A \odot i$;
  end for
end if

Thus far we have completed the descriptions of our multi-radio channel-hopping design. Algorithm 1 provides the pseudocode for the proposed channel-hopping scheme (CHS) operations carried out by a node with w radio interfaces and p orthogonal channels available in the system.

3.4 Mathematical Analysis

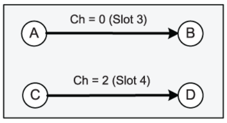
In this section, we analyze the mathematical properties of the proposed channel-hopping scheme (CHS) and prove that network connectivity is well preserved under this scheme. Given p orthogonal channels with p prime, there are $(p+1)$ slots (including the parity slot at Slot 0) contained in a cycle. Below we start with the single-radio system and then extend the analysis to multi-radio channel-hopping design.

3.4.1 Connectivity Properties of Single-radio CHS

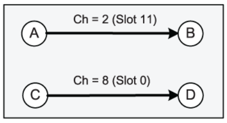
For any two neighboring nodes A and B each equipped with a single radio, three cases are possible for the relationship between their channel schedules. The three cases are $\alpha^A = \alpha^B$ and $\chi^A = \chi^B$, $\alpha^A = \alpha^B$ but $\chi^A \ne \chi^B$, and $\alpha^A \ne \alpha^B$ (whether $\chi^A = \chi^B$ or $\chi^A \ne \chi^B$). We prove that the proposed single-radio CHS is able to preserve network connectivity in all three cases.

Lemma 1. *When $\alpha^A = \alpha^B$ and $\chi^A = \chi^B$, $(p+1)$ slots overlap in a cycle between nodes A and B.*

Proof. In this case, nodes A and B have *identical* channel-hopping schedule. In addition, parity channel $\beta^A = \beta^B$ because $\beta^A = \alpha^A$, $\beta^B = \alpha^B$, and $\alpha^A = \alpha^B$. According to Eq. (1), when $\alpha^A = \alpha^B$ and $\chi^A = \chi^B$, apparently any channel $\chi^A[k]$ visited at the k^{th} slot by node A overlaps with channel $\chi^B[k]$ visited by node B, leading to all $(p+1)$ slots in a cycle are overlapped between nodes A and B. □



p = 5, ω = 1



p = 11, ω = 1

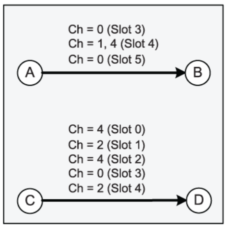
{ β, (SC, Seed) }	Slot / Node	SDU 0	SDE	SDU 1	SDE	SDU 2	SDE	SDU 3	SDE	SDU 4	SDE	SDU 5	SDE
{ 4, (2, 4) }	A	Ch 4		Ch 2		Ch 1		Ch 0		Ch 4		Ch 3	
{ 1, (3, 1) }	B	Ch 1		Ch 3		Ch 4		Ch 0		Ch 1		Ch 2	
{ 4, (0, 4) }	C	Ch 4		Ch 0		Ch 4		Ch 3		Ch 2		Ch 1	
{ 2, (1, 2) }	D	Ch 2		Ch 1		Ch 3		Ch 0		Ch 2		Ch 4	

p = 5, ω = 1

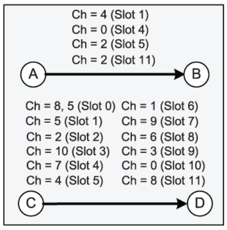
{ β, (SC, Seed) }	Slot / Node	SDU 0	SDE	SDU 1	SDE	SDU 2	SDE	SDU 3	SDE	SDU 4	SDE	SDU 5	SDE	SDU 6	SDE	SDU 7	SDE	SDU 8	SDE	SDU 9	SDE	SDU 10	SDE	SDU 11	SDE
{ 5, (7, 5) }	A	Ch 5		Ch 7		Ch 1		Ch 6		Ch 0		Ch 5		Ch 10		Ch 4		Ch 9		Ch 3		Ch 8		Ch 2	
{ 2, (4, 2) }	B	Ch 2		Ch 4		Ch 6		Ch 8		Ch 10		Ch 1		Ch 3		Ch 5		Ch 7		Ch 9		Ch 0		Ch 2	
{ 8, (5, 8) }	C	Ch 8		Ch 5		Ch 2		Ch 10		Ch 7		Ch 4		Ch 1		Ch 9		Ch 6		Ch 3		Ch 0		Ch 8	
{ 8, (1, 8) }	D	Ch 8		Ch 1		Ch 9		Ch 6		Ch 3		Ch 0		Ch 8		Ch 5		Ch 2		Ch 10		Ch 7		Ch 4	

p = 11, ω = 1

(a)



p = 5, ω = 2



p = 11, ω = 2

{ β, (SC, Seed) }	Slot / Node	SDU 0	SDE	SDU 1	SDE	SDU 2	SDE	SDU 3	SDE	SDU 4	SDE	SDU 5	SDE
{ 4, (2, 4) }	A	Ch 4		Ch 2		Ch 1		Ch 0		Ch 4		Ch 3	
{ 3, (4, 4) }		Ch 3		Ch 4		Ch 3		Ch 2		Ch 1		Ch 0	
{ 1, (3, 1) }	B	Ch 1		Ch 3		Ch 4		Ch 0		Ch 1		Ch 2	
{ 2, (1, 1) }		Ch 2		Ch 1		Ch 2		Ch 3		Ch 4		Ch 0	
{ 4, (0, 4) }	C	Ch 4		Ch 0		Ch 4		Ch 3		Ch 2		Ch 1	
{ 3, (2, 4) }		Ch 3		Ch 2		Ch 1		Ch 0		Ch 4		Ch 3	
{ 2, (1, 2) }	D	Ch 2		Ch 1		Ch 3		Ch 0		Ch 2		Ch 4	
{ 4, (2, 2) }		Ch 4		Ch 2		Ch 4		Ch 1		Ch 3		Ch 0	

p = 5, ω = 2

{ β, (SC, Seed) }	Slot / Node	SDU 0	SDE	SDU 1	SDE	SDU 2	SDE	SDU 3	SDE	SDU 4	SDE	SDU 5	SDE	SDU 6	SDE	SDU 7	SDE	SDU 8	SDE	SDU 9	SDE	SDU 10	SDE	SDU 11	SDE
{ 5, (7, 5) }	A	Ch 5		Ch 7		Ch 1		Ch 6		Ch 0		Ch 5		Ch 10		Ch 4		Ch 9		Ch 3		Ch 8		Ch 2	
{ 10, (4, 5) }		Ch 10		Ch 4		Ch 9		Ch 3		Ch 8		Ch 2		Ch 7		Ch 1		Ch 6		Ch 0		Ch 5		Ch 10	
{ 2, (4, 2) }	B	Ch 2		Ch 4		Ch 6		Ch 8		Ch 10		Ch 1		Ch 3		Ch 5		Ch 7		Ch 9		Ch 0		Ch 2	
{ 4, (5, 2) }		Ch 4		Ch 5		Ch 7		Ch 9		Ch 0		Ch 2		Ch 4		Ch 6		Ch 8		Ch 10		Ch 1		Ch 3	
{ 8, (5, 8) }	C	Ch 8		Ch 5		Ch 2		Ch 10		Ch 7		Ch 4		Ch 1		Ch 9		Ch 6		Ch 3		Ch 0		Ch 8	
{ 5, (9, 8) }		Ch 5		Ch 9		Ch 6		Ch 3		Ch 0		Ch 8		Ch 5		Ch 2		Ch 10		Ch 7		Ch 4		Ch 1	
{ 8, (1, 8) }	D	Ch 8		Ch 1		Ch 9		Ch 6		Ch 3		Ch 0		Ch 8		Ch 5		Ch 2		Ch 10		Ch 7		Ch 4	
{ 5, (5, 8) }		Ch 5		Ch 5		Ch 2		Ch 10		Ch 7		Ch 4		Ch 1		Ch 9		Ch 6		Ch 3		Ch 0		Ch 8	

p = 11, ω = 2

(b)

Fig. 4. Demonstration of high overlapping ratio achieved by our channel-hopping design, where (a) shows the single-radio channel schedules for $p = 5$ and $p = 11$ respectively and (b) displays the channel schedules after adding one more radio and performing the proposed multi-radio channel-hopping scheme (CHS).

Lemma 2. *When $\alpha^A = \alpha^B$ and $\chi^A \neq \chi^B$, there exists one overlapping slot (at the parity slot) in a cycle between nodes A and B.*

Proof. In this case, nodes A and B have *disjoint* channel-hopping schedules and hop in parallel without coming into contact. According to Eq. (1), the parity channel visited at the parity slot is overlapped due to $\alpha^A = \alpha^B$. For non-parity slots in a cycle, suppose there exists an overlapping slot with offset K_{AB} with respect to Slot 1, where $0 \leq K_{AB} < p$. According to Eq. (1), we have $\chi^A \oplus \alpha^A \odot K_{AB} = \chi^B \oplus \alpha^B \odot K_{AB}$. Based on the arithmetic operations of a prime field reviewed in Section 3.1, the overlapping slot offset K_{AB} can be derived as

$$K_{AB} = (\chi^A \oplus (-\chi^B)) \odot (\alpha^B \oplus (-\alpha^A))^{-1}. \tag{3}$$

Since $\alpha^A = \alpha^B$, K_{AB} does not exist. In other words, no overlapping slot exists among non-parity slots. As a result, when $\alpha^A = \alpha^B$ and $\chi^A \neq \chi^B$, one overlapping slot (at the parity slot) exists in a cycle. □

Lemma 3. *When $\alpha^A \neq \alpha^B$, there exists one overlapping slot (at non-parity slot) in a cycle between nodes A and B.*

Proof. According to Eq. (1), the parity channels visited at the parity slot by nodes A and B are different because $\alpha^A \neq \alpha^B$. For non-parity slots in a cycle, suppose there exists an overlapping slot with offset K_{AB} with respect to Slot 1, where $0 \leq K_{AB} < p$. Based on similar calculations from Lemma 2, we have $K_{AB} = (\chi^A \oplus (-\chi^B)) \odot (\alpha^B \oplus (-\alpha^A))^{-1}$ which is obtainable since $\alpha^A \neq \alpha^B$ (no matter $\chi^A = \chi^B$ or $\chi^A \neq \chi^B$). In other words, nodes A and B are expected to meet at Slot $(K_{AB} + 1)$ over channel $\chi^A[K_{AB} + 1]$ $(= \chi^B[K_{AB} + 1])$ among non-parity slots per cycle. As a result, when $\alpha^A \neq \alpha^B$, one overlapping slot (at non-parity slot) exists in a cycle between nodes A and B. □

Theorem 1. *Network connectivity is guaranteed in the single-radio CHS.*

Proof. By Lemma 1, Lemma 2, and Lemma 3, there exists at least one overlapping slot in a cycle between any two neighboring nodes, leading to a guaranteed network connectivity produced by the proposed single-radio channel-hopping scheme (CHS). □

3.4.2 Connectivity Properties of Multi-radio CHS

For any two neighboring nodes A and B with w radios available $(1 < w \leq p)$, four cases are analyzed separately.

These cases include $\alpha_1^A = \alpha_1^B$ and $\chi_1^A = \chi_1^B$, $\alpha_1^A = \alpha_1^B$ but $\chi_1^A \neq \chi_1^B$, $\alpha_1^A \neq \alpha_1^B$ with $\alpha_1^A \neq (-\alpha_1^B)$, and $\alpha_1^A \neq \alpha_1^B$ with $\alpha_1^A = (-\alpha_1^B)$. We prove that the proposed multi-radio CHS is capable of preserving network connectivity in all cases.

Lemma 4. *When $\alpha_1^A = \alpha_1^B$ and $\chi_1^A = \chi_1^B$, there are $(p+1)$ overlapping channels in a cycle between nodes A and B.*

Proof. Based on the multi-radio CHS design, if $\alpha_1^A = \alpha_1^B$ and $\chi_1^A = \chi_1^B$, then $\alpha_i^A = \alpha_i^B$ and $\chi_i^A = \chi_i^B$ for interfaces I_i^A and I_i^B equipped at nodes A and B respectively, where $1 \leq i \leq w$. In other words, for any index i, I_i^A and I_i^B adopt an identical channel-hopping schedule. According to Eq. (2), apparently any channel $\chi_i^A[k]$ visited at the k^{th} slot by interface I_i^A overlaps with channel $\chi_i^B[k]$ visited by interface I_i^B, leading to all $(p+1)$ slots in a cycle are overlapped between nodes A and B. □

Lemma 5. *When $\alpha_1^A = \alpha_1^B$ and $\chi_1^A \neq \chi_1^B$, there exists at least one overlapping slot (at the parity slot) in a cycle between nodes A and B.*

Proof. Based on the multi-radio CHS design, if $\alpha_1^A = \alpha_1^B$ and $\chi_1^A \neq \chi_1^B$, then $\alpha_i^A = \alpha_i^B$ and $\chi_i^A \neq \chi_i^B$ for interfaces I_i^A and I_i^B equipped at nodes A and B, where $1 \leq i \leq w$. According to Eq. (2), the parity channel visited at the parity slot by interfaces I_i^A and I_i^B is the same due to $\alpha_1^A = \alpha_1^B$. For non-parity slots in a cycle, suppose there exists an overlapping slot with offset $K_{A^iB^j}$ with respect to Slot 1 produced by interfaces I_i^A and I_i^B, where $0 \leq K_{A^iB^j} < p$. According to Eq. (2), we have

$$\begin{aligned}&\chi_1^A \oplus \alpha_1^A \odot (\lceil p/w \rceil \odot (i-1) \oplus K_{A^iB^j}) \\ &\quad = \chi_1^B \oplus \alpha_1^B \odot (\lceil p/w \rceil \odot (j-1) \oplus K_{A^iB^j}).\end{aligned} \tag{4}$$

Since $\alpha_1^A = \alpha_1^B$ and $\chi_1^A \neq \chi_1^B$, Eq. (4) does not hold if $i = j$, and holds if $j - i = (\chi_1^A \oplus (-\chi_1^B)) \odot (\alpha_1^A \odot \lceil p/w \rceil)^{-1}$. In other words, no overlapping slot exists among non-parity slots between interfaces I_i^A and I_i^B, but non-parity overlapping occurs between interfaces I_i^A and I_j^B if $j - i = (\chi_1^A \oplus (-\chi_1^B)) \odot (\alpha_1^A \odot \lceil p/w \rceil)^{-1}$. As a result, when $\alpha_1^A = \alpha_1^B$ and $\chi_1^A \neq \chi_1^B$, at least one overlapping slot (at the parity slot) exists in a cycle between nodes A and B. □

Lemma 6. *When $\alpha_1^A \neq \alpha_1^B$ with $\alpha_1^A \neq (-\alpha_1^B)$, there exists at least one and at most $min(p, w^2)$ overlapping slots among non-parity slots in a cycle between nodes A and B.*

Proof. For the first interfaces I_1^A and I_1^B, based on Lemma 3, when $\alpha_1^A \neq \alpha_1^B$, one overlapping among non-parity slots in a cycle is guaranteed. For other overlapping opportunities, since $\alpha_1^A \neq \alpha_1^B$, from Eq. (4), we can derive

$$\begin{aligned}K_{A^iB^j} = &[\chi_1^A \oplus (-\chi_1^B) \\ &\oplus \alpha_1^A \odot (\lceil p/w \rceil \odot (i-1)) \oplus (-\alpha_1^B) \odot (\lceil p/w \rceil \\ &\odot (j-1))] \odot (\alpha_1^B \oplus (-\alpha_1^A))^{-1}.\end{aligned} \tag{5}$$

In other words, for any (i, j) pair, the overlapping offset $K_{A^iB^j}$ produced by interfaces I_i^A and I_j^B, where $1 \leq i, j \leq w$, is obtainable with at most $min(p, w^2)$ distinct values. As a result, when $\alpha_1^A \neq \alpha_1^B$ with $\alpha_1^A \neq (-\alpha_1^B)$, at least one and at most $min(p, w^2)$ non-parity slots are overlapped in a cycle between nodes A and B. □

Lemma 7. *When $\alpha_1^A \neq \alpha_1^B$ with $\alpha_1^A = (-\alpha_1^B)$, there exists at least one and at most $min(p, \frac{w^2+w}{2})$ overlapping slots among non-parity slots in a cycle between nodes A and B.*

Proof. For the first interfaces I_1^A and I_1^B, based on Lemma 3, when $\alpha_1^A \neq \alpha_1^B$, one overlapping among non-parity slots in a cycle is guaranteed. For other overlapping opportunities, since $\alpha_1^A \neq \alpha_1^B$, we also obtain Eq. (5) as in Lemma 6. However, since $\alpha_1^A = (-\alpha_1^B)$ in this case, we can further reduce Eq. (5) into the following formula,

$$K_{A^iB^j} = [\chi_1^A \oplus (-\chi_1^B) \oplus \alpha_1^A \odot (\lceil p/w \rceil \odot (i+j-2))] \odot (2\alpha_1^B)^{-1}. \tag{6}$$

Based on Eq. (6), we observe that $K_{A^iB^j} = K_{A^jB^i}$ due to the fact $i + j = j + i$. In other words, the overlapping produced by interfaces I_i^A and I_j^B is exactly the same as that produced by interfaces I_j^A and I_i^B, leading to at most $min(p, \frac{w^2+w}{2})$ overlapping non-parity slots. As a result, when $\alpha_1^A \neq \alpha_1^B$ with $\alpha_1^A = (-\alpha_1^B)$, at least one and at most $min(p, \frac{w^2+w}{2})$ non-parity slots are overlapped in a cycle between nodes A and B. □

Corollary 1. *From Lemma 6 and Lemma 7, suppose the best case of $min(p, w^2)$ slots overlapping among non-parity slots occurs in a cycle, then the required number of radio interfaces is upper bounded by $\sqrt{p}$ under the CHS design.*

Proof. It follows that $w^2 \leq p$, resulting in $w \leq \sqrt{p}$. □

Theorem 2. *Network connectivity is guaranteed in the multi-radio CHS.*

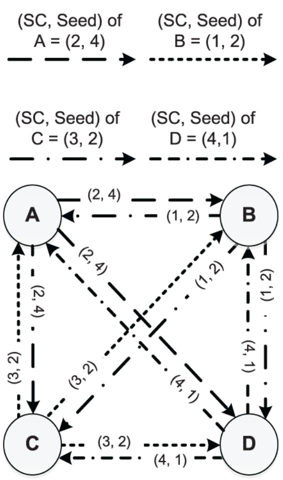
Proof. By Lemma 4, Lemma 5, Lemma 6, and Lemma 7, there exists at least one overlapping slot in a cycle between any two neighboring nodes, leading to a guaranteed network connectivity produced by the proposed multi-radio channel-hopping scheme (CHS). □

4 Channel-Diverse Routing (CDR)

Once channel-hopping schedules are determined, our channel-diverse routing (CDR) comes into play. Several aspects are involved in the CDR protocol. In Section 4.1, we make clear how control packets can be exchanged to gather network information in such channel-hopping environment without a dedicated radio interface for this kind communications traffic. Section 4.2 elaborates on the route discovery procedure, while Section 4.3 presents the technique of managing data queues to further improve delivery efficiency. We summarize the CDR protocol in Section 4.4.

4.1 Network Information Construction

In typical channel-hopping schemes (with distinct hopping schedules), channel diversity can be better exploited but transmitting broadcast packets becomes difficult/complicated due to the lack of a common communications rendezvous [3]. Observing this, we propose to set aside a control channel (CC) and add a Broadcast Slot at Slot $(p+1)$, extending a cycle to include $(p+2)$ time slots, as shown in Fig. 5.



{ β, (SC, Seed) }	Node \ Slot	0	1	2	3	4	5	6
{ 4, (2, 4) }	A	Ch 4	Ch 2	Ch 1	Ch 0	Ch 4	Ch 3	CC
{ 3, (4, 4) }	A	Ch 3	Ch 4	Ch 3	Ch 2	Ch 1	Ch 0	Prob
{ 2, (1, 2) }	B	Ch 2	Ch 1	Ch 3	Ch 0	Ch 2	Ch 4	CC
{ 4, (2, 2) }	B	Ch 4	Ch 2	Ch 4	Ch 1	Ch 3	Ch 0	Prob
{ 2, (3, 2) }	C	Ch 2	Ch 3	Ch 0	Ch 2	Ch 4	Ch 1	CC
{ 4, (4, 2) }	C	Ch 4	Ch 4	Ch 1	Ch 3	Ch 0	Ch 2	Prob
{ 1, (4, 1) }	D	Ch 1	Ch 4	Ch 0	Ch 1	Ch 2	Ch 3	CC
{ 2, (2, 1) }	D	Ch 2	Ch 2	Ch 3	Ch 4	Ch 0	Ch 1	Prob

Channel-hopping Schedule; Slot 0: Parity Slot; Slot 6: Broadcast Slot

Neighbor Table

ID	SC	Seed	Overlapping Channel Set						
			Slot 0	Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6
B	1	2	{4}	{2}	{3}	{0}	Ø	{0}	Ø
C	3	2	{4}	{4}	{1}	{2}	{4}	Ø	Ø
D	4	1	Ø	{2, 4}	{3}	Ø	Ø	{3}	Ø
A	2	4	{4}	{2}	{3}	{0}	Ø	{0}	Ø
C	3	2	{2, 4}	Ø	Ø	Ø	Ø	Ø	Ø
D	4	1	{2}	{2}	{3}	{1}	{2}	Ø	Ø
A	2	4	{4}	{4}	{1}	{2}	{4}	Ø	Ø
B	1	2	{2, 4}	Ø	Ø	Ø	Ø	Ø	Ø
D	4	1	{2}	{4}	{0}	Ø	{0}	{1}	Ø
A	2	4	Ø	{2, 4}	{3}	Ø	Ø	{3}	Ø
B	1	2	{2}	{2}	{3}	{1}	{2}	Ø	Ø
C	3	2	{2}	{4}	{0}	Ø	{0}	{1}	Ø

Fig. 5. Channel-hopping schedule exchange procedure performed during the Broadcast Slot over the control channel (CC) and the established neighbor table at each node. Note that only the first (SC, Seed) pair of each node needs to be broadcast.

Broadcast messages, such as ad hoc beacons, time synchronization signaling, route discovery packets (RREQ/RREP), and channel-hopping schedule exchanges can be performed during the Broadcast Slot over the common control channel (CC). In addition, channel quality probing[3] based on broadcasting back-to-back short packets over data channels (DCs) can also be executed during the Broadcast Slot. If a single radio is equipped at each node, we divide the Broadcast Slot into two mini-slots, each occupying half a slot time. Broadcast messages are sent over CC in the first half of Broadcast Slot, while probing packets are transmitted over some DC in the second half. Given p data channels (DCs) available, it takes p cycles to complete the probing on all DCs. The probing will continue and keep updating current channel qualities over respective DCs. On the other hand, if multiple radios are available, we propose to dedicate the first radio interface for broadcast transmissions over CC and the remaining interface(s) for probing over DCs during the Broadcast Slot. Since two DC probings can be performed in a single Broadcast Slot, with w radios available, in average the probing over all DCs requires $\frac{p}{2(w-1)}$ cycles to be completed. For instance, in Fig. 5, with two radios and five DCs, it takes 2.5 cycles to go through all DC probings. Note that the CC is not designed for the purpose of data channel negotiations, and the usage of CC here provides a common rendezvous for control packets, which is beneficial in facilitating necessary broadcast transmissions demanded by practical network operations.

Fig. 5 depicts the channel-hopping schedule exchange scenario in a local neighborhood between four nodes, whose hopping schedules have been configured according to the proposed channel-hopping scheme (CHS). Two radios are available at each node. Since CHS configures the remaining radios based on the first interface, only the (SC, Seed) pair of the first radio is required to be announced. In other words, *the schedule exchange message set out by each node is fixed in length*, independent of available radio interfaces. This is a nice property in terms of saving communications resource. Once receiving the first (SC, Seed) pair, a node can infer the remaining (SC, Seed) pair(s) for other interface(s) based on the same CHS algorithm. Then a complete neighbor table recording communications channels with a certain neighbor at respective time slots can be constructed, as shown in Fig. 5. Define N_i^A as the set of communicable neighbors for node A at the i^{th} slot, and O_i^{AB} as the set of overlapping channels for nodes A and B at the i^{th} slot. In our example, $N_i^A = \{B, C, D\}$, while $O_0^{AB} = \{4\}$, $O_1^{AB} = \{2\}$, $O_2^{AB} = \{3\}$, $O_3^{AB} = \{0\}$, $O_4^{AB} = \phi$, $O_5^{AB} = \{0\}$, and $O_6^{AB} = \phi$. Some nodes may have multiple communications channels at a certain slot. For instance, $O_1^{AD} = \{2, 4\}$ between nodes A and D at Slot 1. The neighbor tables are updated according to received schedule exchange messages and maintained locally to facilitate route discovery and data delivery.

4.2 Routing Algorithm

Given the constructed network information, we now describe the route discovery procedure. Normal route request (RREQ) and reply (RREP) handshaking process found in most on-demand ad hoc routing protocols, such as DSR and AODV [13], [21], is adopted. We modify a typical DSR RREQ packet to carry extra information helpful in determining a best route from source node S to destination node D. By extra information, we propose to have every node estimate the expected transmission time (ETT) from node S to itself. The metric ETT involves both channel qualities (rates) and overlapping conditions from hop to hop. Although packets can arrive at any time instance (over a continuous time domain), we regard those arriving after the beginning of Slot $i-1$ but before Slot i as packets arriving at Slot i, where $0 \leq i \leq p+1$. Consequently, the ETT information is represented by a vector denoted as $ETT[0], ETT[1], \ldots, ETT[p+1]$ to indicate expected transmission times when packets at source node S emerge at Slot 0, Slot 1, ..., and Slot $(p+1)$ respectively. Initially, node S sets $ETT[i] = 0$ for $i = 0, 1, \ldots, p+1$. Suppose node A receives the RREQ from node S, node A checks its neighbor table on overlapping channel sets with node S at respective time slots.

3. Each node performs the channel quality probing to estimate data rates for neighboring links over all DCs.

If the overlapping channel set is non-empty at Slot i, then the estimated average link transmission time (LTT) is added to $ETT[i]$. Otherwise, the transmission needs to be postponed for, say, k slots, then the slot waiting time $k(t_{SDU}+t_{SDE})$ should also be counted into the calculation of $ETT[i]$. As the RREQ propagates further, each node should first estimate the arrival slot based on current $ETT[i]$ carried by the received RREQ. Specifically, suppose node B receives the RREQ from node A, based on $ETT[i]$, node B is able to estimate the arrival slot j by computing

$$j = \left(i + \left\lfloor \frac{ETT[i]}{t_{SDU}+t_{SDE}} \right\rfloor \right) \ mod \ \ (p+2). \tag{7}$$

Then node B checks its neighbor table on the overlapping channel set (O_j^{AB}) with node A at Slot j. In case O_j^{AB} is non-empty, node B simply adds link transmission time (LTT) to $ETT[i]$. Otherwise, the transmission need be postponed and slot waiting time should be counted in. Algorithm 2 provides the pseudocode on RREQ forwarding mechanism and detailed ETT vector calculations exercised by CDR protocol. Based on the ETT vector carried in RREQ packet, the destination node D obtains the average ETT by calculating $\frac{\sum_{i=0}^{p+1} ETT[i]}{p+2}$, assuming equal probability for packets to arrive at any slot at node S. After gathering three RREQ packets or timeout expires, node D determines a best route with the minimum average ETT value and sends RREP back to node S along the selected route.

Algorithm 2 Forwarding Mechanism When a New RREQ Has Been Received

Suppose this new RREQ is originated from node S and received by node B from node A;
// initially $ETT[i]=0$ for $i=0,1,2,\ldots,p+1$ which are carried in the RREQ issued by node S
for ($i=0$; $i \le p+1$; $i{+}{+}$) **do**
  Set $j = (i + \left\lfloor \frac{ETT[i]}{(t_{SDU}+t_{SDE})} \right\rfloor)\ mod\ (p+2)$; // arrival slot
  Set $r = ETT[i]\ mod\ (t_{SDU}+t_{SDE})$;
  if ($O_j^{AB} \neq \phi$) **then**
    Set $ETT[i] = ETT[i] + LTT_{avg}$;
    // link transmission time (LTT) averaged from all communicable data channels in O_j^{AB}
  else
    Set $k=0$;
    while ($O_j^{AB} == \phi$) **do**
      Set $j=(j+1)\ mod\ (p+2)$;
      Set $k=k+1$;
    end while
    Set $ETT[i] = ETT[i] - r + k(t_{SDU}+t_{SDE}) + LTT_{avg}$;
  end if
end for
Append the address of node B in RREQ;
Update the ETT vector ($ETT[0], ETT[1], \ldots, ETT[p+1]$) in RREQ;
Rebroadcast the modified RREQ packet;

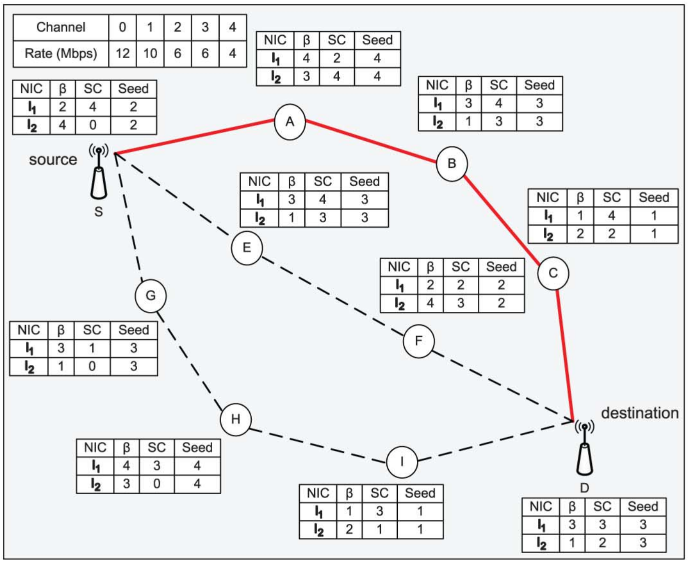
Fig. 6 displays a CDR route discovery example in a multi-hop network with $p=5$ and $w=2$. Three routes are discovered: $\{S,A,B,C,D\}$, $\{S,E,F,D\}$, and $\{S,G,H,I,D\}$. Route $\{S,E,F,D\}$ is the shortest but the overlapping slots between nodes S and E concentrate at Slot 1, 2, and 3, leading to significantly increased ETT values in $RREQ_{SE}$, which affects the resultant ETT values obtained by destination node D. Similar problem occurs in route $\{S,G,H,I,D\}$ between nodes H and I. As a result, route $\{S,A,B,C,D\}$ with the minimum average ETT value is selected. This demonstrates the route selection mechanism exercised by the proposed CDR, which successfully quantifies both the channel qualities (rates) and overlapping conditions in a unified ETT calculation model[4].

4.3 Data Queue Management

In order to distinguish the purpose of link-layer packets, each node normally keeps a broadcast queue and a (unicast) data queue. Broadcast packets, such as ad hoc beacons, timing messages, probings, and channel schedule exchanges, are scheduled periodically and served in order. Likewise, traditionally packets in data queue are enqueued in the order of their arrival times and dequeued (served) in a first-in first-out (FIFO) manner. However, in channel-hopping wireless environments, such single data queue approach can lead to significantly extended transmission processing time. For example, in Fig. 7, three flows go through node S: $\{S,A,C\}$ (Flow 1), $\{S,B,D\}$ (Flow 2), and $\{S,B,E\}$ (Flow 3). Two packets destined for node C arrive before packets destined for nodes D and E as illustrated in the figure at time t_0 (the beginning of Slot 0). When single (data) queue approach is used, node S has to wait until Slot 5 for delivering the head-of-line (HOL) packet C despite that packets D and E can be delivered earlier at Slot 2. Similar problem occurs at node B when packet E (which can be delivered at Slot 3) is blocked by packet D (which can be delivered later at Slot 4).

In light of this, we propose to implement per-destination queues to avoid blocking transmittable packets due to a HOL packet whose receiver is currently out of reach. As illustrated in Fig. 7 (lower right), node S maintains queues q_C, q_D, and q_E for destination nodes C, D, and E separately. For queue services, we perform selective round-robin (SRR) mechanism that works as follows. Recall that N_i^S is the set of communicable neighbors for node S at the i^{th} slot. Denote Q_i^S as the set of *serviceable destination queues* at the i^{th} slot for node S. By serviceable destination queues, we mean those queues whose next-hop nodes to their respective destinations are in set N_i^S. During each slot time, node S establishes set Q_i^S and serves the selected queues in a round-robin manner. For each queue in service, the first-in first-out (FIFO) rule is applied. Algorithm 3 provides the pseudocode for the SRR mechanism performed

4. Once a packet is sent over a certain wireless channel, the transmitter-receiver pair will not perform channel switching (even when the next time slot begins) until the packet is completely processed.



Slot / Node	SDU 0	SDU 1	SDU 2	SDU 3	SDU 4	SDU 5	SDU 6
S	Ch 2	Ch 4	Ch 1	Ch 3	Ch 0	Ch 2	CC
	Ch 4	Ch 0	Ch 2	Ch 4	Ch 1	Ch 3	Prob
A	Ch 4	Ch 2	Ch 1	Ch 0	Ch 4	Ch 3	CC
	Ch 3	Ch 4	Ch 3	Ch 2	Ch 1	Ch 0	Prob
B	Ch 3	Ch 4	Ch 2	Ch 0	Ch 3	Ch 1	CC
	Ch 1	Ch 3	Ch 1	Ch 4	Ch 2	Ch 0	Prob
C	Ch 1	Ch 4	Ch 0	Ch 1	Ch 2	Ch 3	CC
	Ch 2	Ch 2	Ch 3	Ch 4	Ch 0	Ch 1	Prob
D	Ch 3	Ch 3	Ch 1	Ch 4	Ch 2	Ch 0	CC
	Ch 1	Ch 2	Ch 0	Ch 3	Ch 1	Ch 4	Prob

Slot / Node	SDU 0	SDU 1	SDU 2	SDU 3	SDU 4	SDU 5	SDU 6
S	Ch 2	Ch 4	Ch 1	Ch 3	Ch 0	Ch 2	CC
	Ch 4	Ch 0	Ch 2	Ch 4	Ch 1	Ch 3	Prob
E	Ch 3	Ch 4	Ch 2	Ch 0	Ch 3	Ch 1	CC
	Ch 1	Ch 3	Ch 1	Ch 4	Ch 2	Ch 0	Prob
F	Ch 2	Ch 2	Ch 4	Ch 1	Ch 3	Ch 0	CC
	Ch 4	Ch 3	Ch 0	Ch 2	Ch 4	Ch 1	Prob
D	Ch 3	Ch 3	Ch 1	Ch 4	Ch 2	Ch 0	CC
	Ch 1	Ch 2	Ch 0	Ch 3	Ch 1	Ch 4	Prob

Slot / Node	SDU 0	SDU 1	SDU 2	SDU 3	SDU 4	SDU 5	SDU 6
S	Ch 2	Ch 4	Ch 1	Ch 3	Ch 0	Ch 2	CC
	Ch 4	Ch 0	Ch 2	Ch 4	Ch 1	Ch 3	Prob
G	Ch 3	Ch 1	Ch 4	Ch 2	Ch 0	Ch 3	CC
	Ch 1	Ch 0	Ch 3	Ch 1	Ch 4	Ch 2	Prob
H	Ch 4	Ch 3	Ch 2	Ch 1	Ch 0	Ch 4	CC
	Ch 3	Ch 0	Ch 4	Ch 3	Ch 2	Ch 1	Prob
I	Ch 1	Ch 3	Ch 4	Ch 0	Ch 1	Ch 2	CC
	Ch 2	Ch 1	Ch 2	Ch 3	Ch 4	Ch 0	Prob
D	Ch 3	Ch 3	Ch 1	Ch 4	Ch 2	Ch 0	CC
	Ch 1	Ch 2	Ch 0	Ch 3	Ch 1	Ch 4	Prob

RREQ	seq	src	dst	ETT[0]	ETT[1]	ETT[2]	ETT[3]	ETT[4]	ETT[5]	ETT[6]	Route	
$RREQ_S$	1	S	D	0	0	0	0	0	0	0	{S}	
$RREQ_{SA}$	1	S	D	1	1	0.4	11.4	0.4	0.67	12	{S, A}	
$RREQ_{SAB}$	1	S	D	1.67	2	0.8	22.33	11.33	1	12.67	{S, A, B}	
$RREQ_{SABC}$	1	S	D	2.07	3	12	22.73	11.73	1.4	13.07	{S, A, B, C}	
$RREQ_{SABCD}$	1	S	D	2.47	3.67	13	44.4	33.4	22.4	13.47	{S, A, B, C, D}	Selected Route with Average ETT = 18.97 ms
$RREQ_{SE}$	1	S	D	12	1	0.53	1	45	34	23	{S, E}	
$RREQ_{SEF}$	1	S	D	12.67	1.67	22.67	11.67	45.67	34.67	23.67	{S, E, F}	
$RREQ_{SEFD}$	1	S	D	13.34	2.34	33.33	22.33	46.34	35.34	24.34	{S, E, F, D}	Average ETT = 25.34 ms
$RREQ_{SG}$	1	S	D	11.33	0.33	22.33	11.33	0.33	0.67	22.33	{S, G}	
$RREQ_{SGH}$	1	S	D	11.66	0.66	22.66	11.66	0.66	22.67	22.66	{S, G, H}	
$RREQ_{SGHI}$	1	S	D	12.33	1.33	66.67	55.67	44.67	33.67	23.33	{S, G, H, I}	
$RREQ_{SGHID}$	1	S	D	13	2	67.34	56.34	45.34	34.34	24	{S, G, H, I, D}	Average ETT = 34.62 ms

Fig. 6. Example of route selection exercised by the proposed channel-diverse routing (CDR) in a multi-hop wireless network ($p = 5, \omega = 2$). The ETT calculations (in ms) are based on the following settings: packet size $S = 500$ bytes, $t_{SDU} = 10$ ms and $t_{SDE} = 1$ ms.

p = 5, ω = 1 (Flow 1)

{ β, (SC, Seed) }	Slot / Node	SDU 0	SDU 1	SDU 2	SDU 3	SDU 4	SDU 5	SDU 6
{ 2, (4, 2) }	S	Ch 2	Ch 4	Ch 1	Ch 3	Ch 0	Ch 2	CC / Prob
{ 3, (3, 3) }	A	Ch 1	Ch 3	Ch 4	Ch 0	Ch 1	Ch 2	CC / Prob
{ 2, (0, 2) }	C	Ch 4	Ch 3	Ch 2	Ch 1	Ch 0	Ch 4	CC / Prob

p = 5, ω = 1 (Flow 2)

{ β, (SC, Seed) }	Slot / Node	SDU 0	SDU 1	SDU 2	SDU 3	SDU 4	SDU 5	SDU 6
{ 2, (4, 2) }	S	Ch 2	Ch 4	Ch 1	Ch 3	Ch 0	Ch 2	CC / Prob
{ 3, (3, 3) }	B	Ch 3	Ch 3	Ch 1	Ch 4	Ch 2	Ch 0	CC / Prob
{ 2, (0, 2) }	D	Ch 1	Ch 4	Ch 0	Ch 1	Ch 2	Ch 3	CC / Prob

p = 5, ω = 1 (Flow 3)

{ β, (SC, Seed) }	Slot / Node	SDU 0	SDU 1	SDU 2	SDU 3	SDU 4	SDU 5	SDU 6
{ 2, (4, 2) }	S	Ch 2	Ch 4	Ch 1	Ch 3	Ch 0	Ch 2	CC / Prob
{ 3, (3, 3) }	B	Ch 3	Ch 3	Ch 1	Ch 4	Ch 2	Ch 0	CC / Prob
{ 2, (0, 2) }	E	Ch 2	Ch 0	Ch 2	Ch 4	Ch 1	Ch 3	CC / Prob

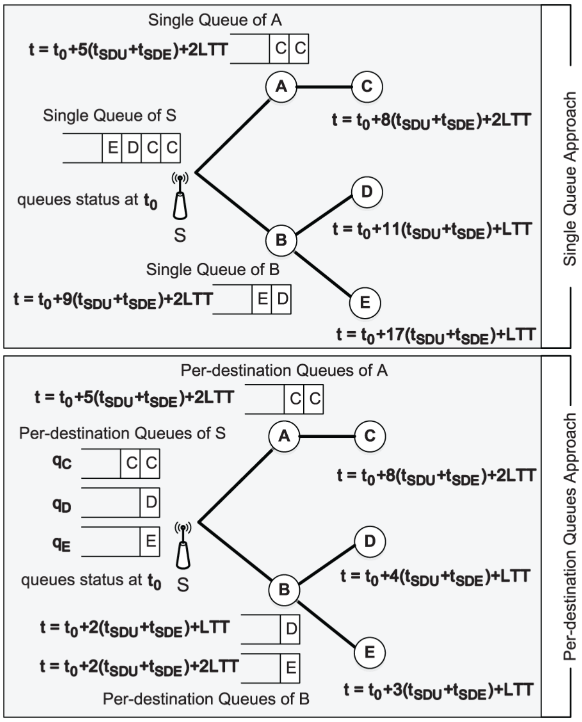


Fig. 7. Advantage of implementing per-destination queues (lower right) to accelerate data delivery by avoiding blocking transmittable packets. During each slot time, deliverable flows (selective destination queues) can be served in a round-robin manner.

TABLE 1
Summary of Notations Used in This Paper

Notation	Description
p	Number of available orthogonal (non-overlapping) channels
w	Number of total radio interfaces equipped at a node, where $w \leq p$
t_{SDU}	Slot duration time
t_{SDE}	Channel switching delay
R_i	Link rate of channel C_i, where $0 \leq i < p$
I_i^A	The i-th radio interface of node A, where $1 \leq i \leq \omega$
χ_i^A	Starting channel configured for interface I_i^A, where $0 \leq \chi_i^A < p$
α_i^A	Channel seed configured for interface I_i^A, where $1 \leq \alpha_i^A < p$
β_i^A	Parity channel configured for interface I_i^A, where $0 \leq \beta_i^A < p$
$K_{A^iB^j}$	Overlapping slot offset with respect to Slot 1, produced by interface I_i^A and interface I_j^B, where $1 \leq i, j \leq w$ and $0 \leq K_{A^iB^j} < p$
N_i^A	Set of communicable neighbors for node A at the i-th slot
O_i^{AB}	Set of overlapping channels for nodes A and B at the i-th slot, where $0 \leq i \leq p+1$
Q_i^A	Set of serviceable destination queues at the i-th slot for node A

during each slot time. Our goal is to transport as many packets as possible in a single time slot. By exercising the proposed data queue management, the delivery times for packets D and E have been effectively reduced by $7(t_{SDU} + t_{SDE})$, equivalent to one cycle time, and $14(t_{SDU} + t_{SDE})$, equal to two cycle time, respectively, as shown in Fig. 7. This queue management skill is essential in realizing efficient data delivery under the channel-hopping scenario, especially for nodes that act as common relaying points and serve multiple destinations.

Algorithm 3 SRR Algorithm Performed during the i-th Slot Time for Node A

```
Construct Q_i^A with supposedly K elements; // denoted as q_0, q_1, ..., q_{K-1}
Set k = 0;
while (! end of the i-th slot time) do
    Look up the next-hop node n_k for the destination queue q_k; // n_k ∈ N_i^A
    if ((∃ idle channel C_j ∈ O_i^{A n_k}) && (q_k not empty)) then
        Pop the head-of-line (HOL) packet; // dequeue operation from q_k
        Complete the DCF contention procedure and transmit over channel C_j;
        // in case multiple idle channels exist, just randomly choose one
    end if
    if (end of the i-th slot time) then
        break;
    end if
    Set k = (k + 1) mod K;
end while
```

4.4 Summary

We have completed the design of channel-diverse routing (CDR) that operates under the channel-hopping scheme (CHS) presented in Section 3. Note that in this time-divided system, there may be ongoing transmissions on slot boundaries. In those cases, CDR lets the transmissions be completed before switching back to predefined hopping schedule. Moreover, the ETT calculations exercised by CDR are based on the given link capacities, which are not easy to obtain precisely in practice. We implement the probing mechanism to estimate link capacities in our simulative experiments, which show that certain inaccuracies do exist but are allowed. As long as the relative channel qualities are not drastically reversed, channel probing is beneficial and the probing overhead can be more than made up for by the increased overall throughput. Finally, we summarize the notations used in this paper in Table 1 for readers' convenience.

5 Performance Evaluation

In this section, simulative experiments are conducted to validate the proposed CDR protocol performance. We extend the ns simulator (version 2.29) to support multi-channel multi-radio (MC-MR) IEEE 802.11a environments. Two-ray ground propagation model is used, and default transmit power (leading to a nominal transmission range of 250 meters) remains unchanged. Table 2 summarizes the parameter settings in our ns simulator. In IEEE 802.11a, eight (up to twelve) orthogonal channels are available, seven of which are used as data channels, denoted as Ch 0, Ch 1, . . ., Ch 6 in the order of decreasing link rates (as shown in Table 2). The remaining channel, Ch 7, is used as the control channel with link rate set at 6 Mbps. For comparison purpose, we implement four existing multi-channel protocols: Static ($w = 2$), Random Static ($w = 2$), FCR-MAC ($w = 2$) [9], and SSCH ($w = 1$) [3], where w indicates the number of radio(s) equipped at each node. The Static approach permanently binds the two radios to Ch 0 and Ch 1 with the highest link rates. DSR RREQs are flooded over the two channels to discover the shortest multi-hop route

TABLE 2
Parameter Settings in Our NS Simulator

Parameter	Value
Simulator version	NS 2.29 with MC-MR extension
Propagation model	Two-ray ground
Transmit power	Default (0.281838 Watt)
Transmission range	250 meters
Number of nodes	9, 16, 25, 36, 49, 64, 81, 100
t_{SDU} / t_{SDE}	10 ms / 1 ms
Packet size	512 bytes
Sending rate	600 Kbps
Simulation time	10 seconds
Channel capacity (Mbps)	Ch 0 (6), Ch 1 (5.4), Ch 2 (4.5), Ch 3 (4.2), Ch 4 (3.6), Ch 5 (3), Ch 6 (1.8), Ch 7 (6)

(with minimum hop count). Using the same routing mechanism as Static, Random Static (RS) organizes the total 8 channels into four sets (each containing 2 channels), and evenly divides the simulation time into four portions. All RS nodes simultaneously switch to the next channel set when a new time portion begins, so that network connectivity can be maintained and all 8 channels can be utilized. The FCR-MAC protocol dedicates one radio to Ch 7 for control traffic and performs an on-demand channel allocation (negotiation) among the other 7 data channels (Ch 0–Ch 6). The negotiation process in FCR-MAC is based on the RTS/CTS handshaking procedure. Once the data channel has been determined, the other radio switches to the selected channel for data delivery. DSR RREQs are propagated over the control channel (Ch 7) to discover the shortest multi-hop path. The SSCH mechanism, similar to our single-radio CHS, is a time-divided channel-hopping scheme. In SSCH, 7 data channels (Ch 0–Ch 6) are utilized to produce two interlacing (channel, seed) pairs that determine the channel-hopping schedule followed by a node. Each (channel, seed) pair in SSCH is a pseudo-random sequence as defined in Section 3.2. Therefore the network connectivity can be guaranteed in SSCH. When there is a communications need, the intended transmitter switches one of its (channel, seed) pair to synchronize with one of the receiver's hopping schedules. Due to a lack of common channel, broadcast transmissions are difficult in SSCH. Necessary broadcast messages, such as ad hoc beacons, timing offsets, channel schedule exchanges, and DSR routing packets need be transmitted over *all* data channels. In the simulations, we set aside 2 ms before every time slot for these broadcast traffic. Unlike SSCH, our CDR protocol uses Ch 7 as the control channel and implements a Broadcast Slot at Slot 8 of each cycle. We simulate CDR ($w = 1$) and CDR ($w = 2$). Both SSCH and CDR employ the time-divided channel-hopping mechanism with slot duration $t_{SDU} = 10$ ms and channel switching delay $t_{SDE} = 1$ ms. In CDR, the destination collects three RREQs or waits until timeout, set at 10 ms, expires, and returns RREP with the minimum average ETT as the best route, which may not be the shortest. On the other hand, SSCH adopts the discovered shortest route with minimum hop count. Except for FCR-MAC, RTS/CTS handshaking is disabled in the simulated multi-channel approaches.

In a simulated network with N wireless nodes (N ranges from 9 to 100) randomly deployed, we arbitrarily generate $\lfloor \frac{N}{4} \rfloor$ CBR traffic flows (proportional to the number of nodes) with per packet size set at 512 bytes and sending rate at 600 Kbps. For a certain number of generated flows, we create 10 random source-destination configurations and obtain an averaged value produced by respective approaches. Fig. 8 plots the performance results in terms of first-bit latency, system throughput, and average end-to-end delay under various network sizes (up to 100 wireless nodes). From this figure, our CDR approach has the first-bit latency lower than SSCH but higher than FCR-MAC and Static methods. Both SSCH and CDR are channel-hopping based schemes, which inherently incur higher first-bit latency because of the longer time required to discover routes for flows. In the case of SSCH, the latency becomes more pronounced due to the lack of a common broadcasting channel for exchanging channel-hopping schedules and other control information between wireless nodes. In contrast, our CDR has a dedicated control channel and broadcast slot for exchanging control information, including channel-hopping schedules, in a timely manner (within one cyclic timeframe). As such, both CDR ($w = 2$) and CDR ($w = 1$) not only require less time than SSCH for the first packet to arrive, but also produce better system throughput than SSCH. On the other hand, although FCR-MAC and Static methods have the least first-bit latency, they do not perform well in terms of system throughput and average delay due to the inappropriate non-quality-aware routes chosen for their flows. In our CDR approach, once the routes are readily available, subsequent flow traffic transmissions become smooth and efficient, leading to the lowest average end-to-end delay despite requiring higher first-bit latency than FCR-MAC and Static methods. Consequently, the results in Fig. 8 demonstrate that both CDR ($w = 2$) and CDR ($w = 1$) outperform the other schemes with the highest system throughput produced, ensuring the CDR protocol designs.

To analyze the protocol behaviors in more detail, we observe a specific network configuration, as depicted in Fig. 9. In the figure, 25 nodes numbered as 0, 1, . . ., 24 and 8 source-destination pairs denoted as S_1/D_1, S_2/D_2, . . ., S_8/D_8 are displayed. The eight flows are generated in order. We observe the aggregate throughput under respective multi-channel

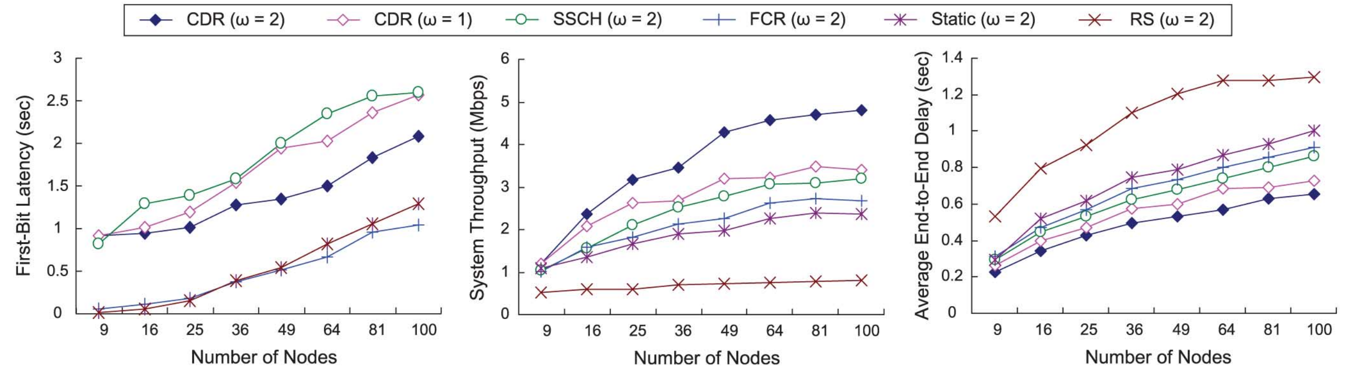
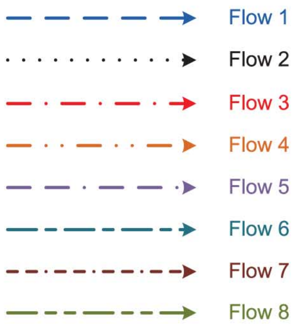


Fig. 8. First-bit latency, system throughput, and average end-to-end delay comparison under static IEEE 802.11a multi-hop wireless networks with number of nodes ranging from 9 to 100.

Channel #	Ch 0	Ch 1	Ch 2	Ch 3	Ch 4	Ch 5	Ch 6
Capacity (Mbps)	6	5.4	4.5	4.2	3.6	3	1.8



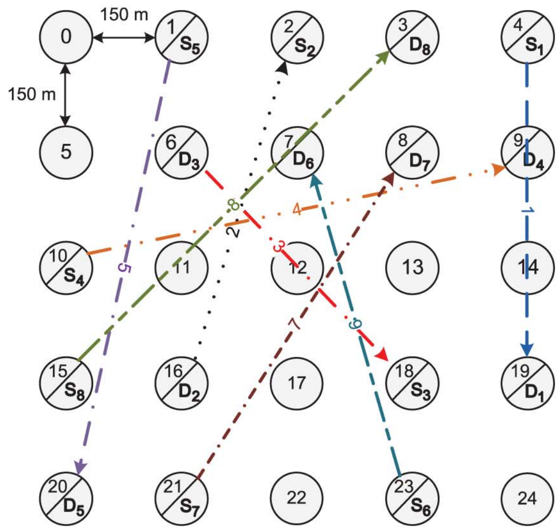


Fig. 9. Network configuration with eight randomly-generated traffic flows.

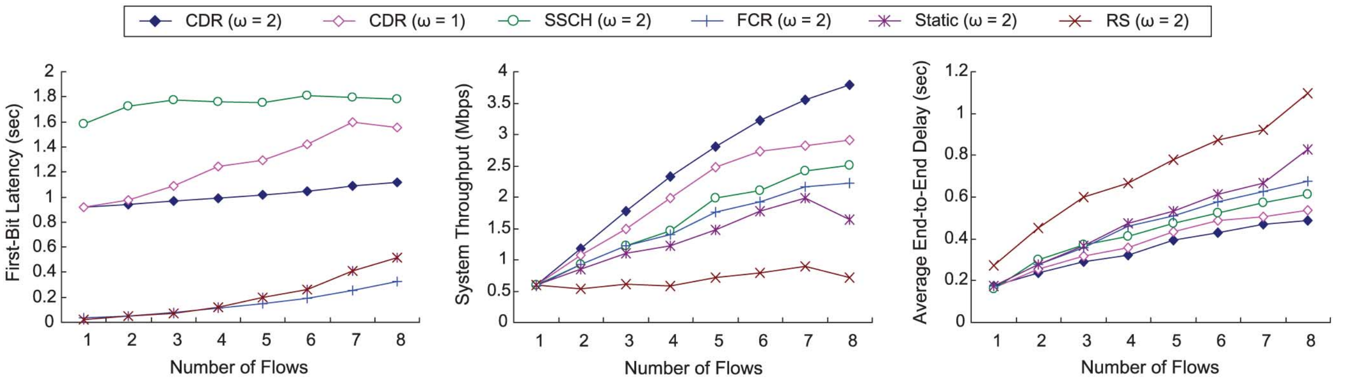


Fig. 10. First-bit latency, system throughput, and average end-to-end delay comparison against number of generated flows in a static 25-node multi-hop wireless network.

approaches as traffic flows saturate the system gradually. Fig. 10 shows the result. In terms of system throughput and average end-to-end delay, our CDR approach outperforms others at all times, while Random Static (RS) produces the lowest system throughput and highest average delay. FCR-MAC and SSCH outperform Static marginally, and the benefit brought by FCR-MAC and SSCH is insufficient to compensate for the overhead they incur. When we observe the first-bit latency in Fig. 10, SSCH requires the longest first-bit arrival time due to its time-consuming discovery of channel-hopping schedules between wireless nodes. Despite not having the shortest first-bit arrival time, our CDR approach yields the best system throughput.

In order to further investigate the resultant performance achieved by respective approaches, we analyze the route distribution and channel usage in Fig. 11. We plot the selected routes for all eight flows and list the channels used by all wireless links. In CDR and SSCH, we also mark the transmission

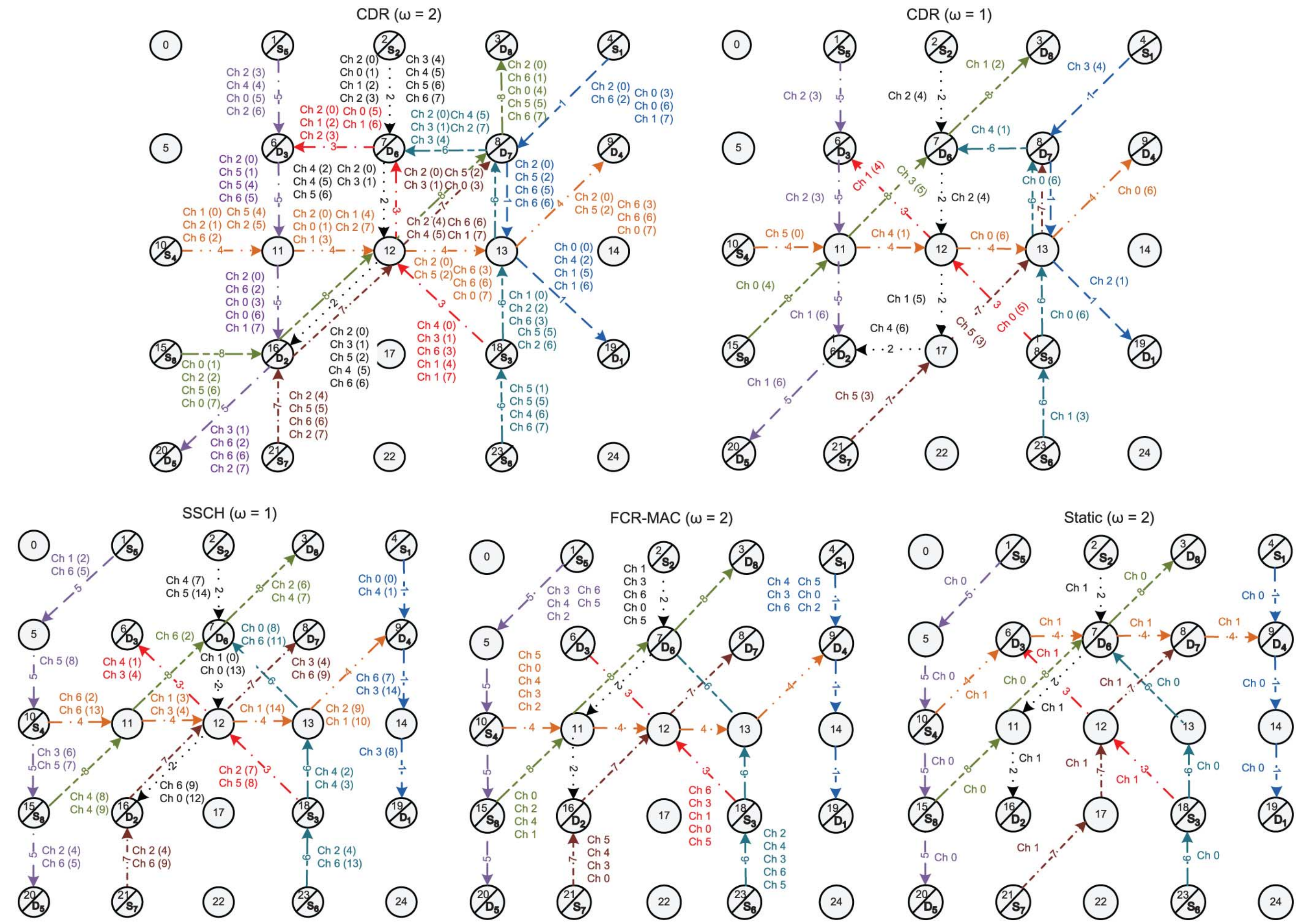
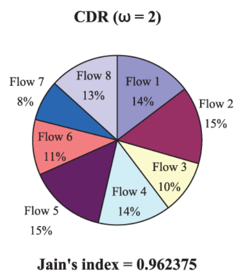
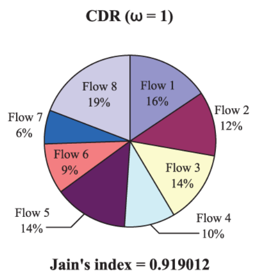


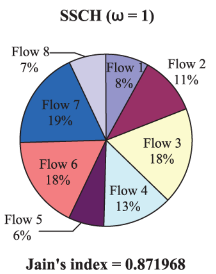
Fig. 11. Route distribution and channel usage under respective multi-channel approaches when eight flows saturate the system.



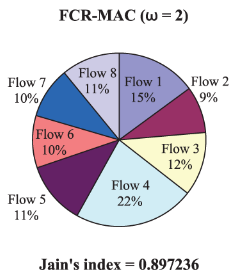
Flow ID	Throughput (Mbps)
1	0.5602
2	0.5520
3	0.3933
4	0.5284
5	0.5617
6	0.3996
7	0.2920
8	0.5064
total	3.7938



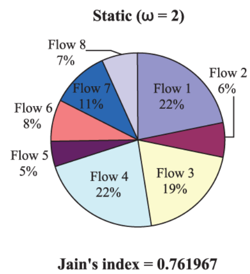
Flow ID	Throughput (Mbps)
1	0.4569
2	0.3514
3	0.3945
4	0.2842
5	0.4037
6	0.2760
7	0.1892
8	0.5579
total	2.9139



Flow ID	Throughput (Mbps)
1	0.2040
2	0.2749
3	0.4598
4	0.3277
5	0.1631
6	0.4409
7	0.4663
8	0.1768
total	2.5135



Flow ID	Throughput (Mbps)
1	0.3332
2	0.1905
3	0.2651
4	0.5019
5	0.2552
6	0.2160
7	0.2160
8	0.2416
total	2.2195



Flow ID	Throughput (Mbps)
1	0.3604
2	0.1026
3	0.3178
4	0.3677
5	0.0796
6	0.1234
7	0.1795
8	0.1097
total	1.6408

Fig. 12. Analysis of per-flow throughput under respective multi-channel approaches.

slots in parentheses. For example, Ch 2 (0) indicates the transmission can be performed at Slot 0 over Ch 2. As we can see from the figure, CDR does not always select the shortest routes, such as flows 3, 6, and 8. Even with the same number of hops, CDR tends to travel through channel-diverse routes to mitigate both intra-flow and inter-flow interferences. This channel-diversity is beneficial as well in balancing throughput share among flows, as illustrated in Fig. 12. We use Jain's fairness index as defined by $\frac{(\sum_{i=1}^{n} X_i)^2}{n \times \sum_{i=1}^{n} X_i^2}$, where n is the number of flows and X_i is the throughput of flow i. Among all simulated approaches, CDR is able to maintain the highest fairness while producing the most system throughput. Three main features of CDR attribute to this result. First, the design of CHS exploits channel randomness with guaranteed network connectivity (no logical partition) and makes sure overlapping slots are evenly distributed. Second, the corresponding CDR protocol is channel quality aware, taking care of link rates and overlapping conditions along the route. Third, our data queue management mechanism avoids blocking deliverable packets, such that a relaying node can serve as many destinations as possible in a single time slot. When traffic is heavy and multiple flows go through the same relaying node, our SRR scheme nicely handles traffic demands without wasting slot time.

6 Conclusion

In this paper, we proposed a connectivity-preserving multi-radio channel-hopping scheme (CHS) and its corresponding channel-diverse routing (CDR) protocol for static multi-hop wireless environments with multiple orthogonal channels available. Most prior multi-channel protocols focus on the channel allocation mechanism, while leaving the multi-hop routing issue unaddressed. We observed the two issues should be jointly considered, and thus designed our multi-channel protocol suite to include both the channel allocation (CHS) and multi-hop routing (CDR) mechanisms. Simulation results have shown that the proposed approach effectively improved system throughput and achieved better balanced throughput share among flows than other simulated multi-channel protocols in a noticeable way.

Acknowledgments

This research was co-sponsored in part by the NSC of Taiwan under Grant 102-2221-E-009-014, and in part by the MoE Program Aiming for the Top University and Elite Research Center Development Plan (ATU Plan).

References

[1] Maxim Integrated Inc. (2013). "Maxim 2.4 GHz 802.11b Zero-IF Transceivers," [Online]. Available: http://pdfserv.maxim-ic.com/en/ds/MAX2820-MAX2821A.pdf.

[2] M. Alicherry, R. Bhatia, and L. Li, "Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks," in *Proc. ACM Annu. Int. Conf. Mobile Comput. Netw. (Mobicom)*, Sep. 2005.

[3] 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 Annu. Int. Conf. Mobile Comput. Netw. (MobiCom)*, Sep. 2004.

[4] K. Bian, J.-M. J. Park, and R. Chen, "A quorum-based framework for establishing control channels in dynamic spectrum access networks," in *Proc. ACM Annu. Int. Conf. Mobile Comput. Netw. (MobiCom)*, Sep. 2009, pp. 25–36.

[5] L.-J. Chen, T. Sun, G. Yang, M. Y. Sanadidi, and M. Gerla, "Ad hoc probe: Path capacity probing in wireless ad hoc networks," in *Proc. IEEE Int. Wireless Internet Conf. (WICON)*, Jul. 2005, pp. 156–163.

[6] A. K. Das, R. Vijayakumar, and S. Roy, "Static channel assignment in multi-radio multi-channel 802.11 wireless mesh networks: Issues, metrics and algorithms," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, Nov. 2006.

[7] D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," *ACM Wireless Netw.*, vol. 11, no. 4, pp. 419–434, Jul. 2005.

[8] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts," in *Proc. USENIX Symp. Oper. Syst. Design Implementation (OSDI)*, Dec. 2002.

[9] M. D. Felice, G. Zhu, and L. Bononi, "Future channel reservation medium access control (FCR-MAC) protocol for multi-radio multi-channel wireless mesh networks," in *Proc. ACM Symp. Perform. Eval. Wireless Ad Hoc Sens. Ubiquitous Netw. (PE-WASUN)*, Oct. 2008.

[10] F. Herzel, G. Fischer, and H. Gustat, "An integrated CMOS RF synthesizer for 802.11a wireless LAN," *IEEE J. Solid-State Circuits*, vol. 38, no. 10, pp. 1767–1770, Oct. 2003.

[11] J. M. Howie, *Fields and Galois Theory [electronic resource]*. New York, NY, USA: Springer-Verlag, 2006.

[12] R. Huang, H. Zhai, C. Zhang, and Y. Fang, "SAM-MAC: An efficient channel assignment scheme for multi-channel ad hoc networks," *Elsevier Comput. Netw.*, vol. 52, no. 8, pp. 1634–1646, Jun. 2008.

[13] D. B. Johnson, Y.-C. Hu, and D. A. Maltz, "The dynamic source routing protocol (DSR) for mobile ad hoc wireless networks for IPv4," *IETF RFC 4728*, 2007.

[14] P. Kyasanur and N. H. Vaidya, "Capacity of multi-channel wireless networks: Impact of number of channels and interfaces," in *Proc. ACM Annu. Int. Conf. Mobile Comput. Netw. (MobiCom)*, Aug. 2005.

[15] P. Kyasanur and N. H. Vaidya, "Routing and link-layer protocols for multi-channel multi-interface ad hoc wireless networks," *Mobile Comput. Commun. Rev.*, vol. 10, no. 1, pp. 31–43, Jan. 2006.

[16] E.-K. Lee, S. Y. Oh, and M. Gerla, "Randomized channel hopping scheme for anti-jamming communication," in *Proc. IFIP Wireless Days (WD)*, Oct. 2010, pp. 1–5.

[17] T.-Y. Lin, W.-H. Tam, K.-L. Fan, and Y.-C. Tseng, "Resource planning and packet forwarding in multi-radio, multi-mode, multi-channel, multi-rate (M4) wireless mesh networks," *Elsevier Comput. Commun.*, vol. 31, no. 7, pp. 1329–1342, May 2008.

[18] B. F. Lo, "A survey of common control channel design in cognitive radio networks," *Elsevier Ad Hoc Netw.*, vol. 8, no. 4, pp. 430–438, Jun. 2010.

[19] J. Mo, H.-S. W. So, and J. Walrand, "Comparison of multichannel MAC protocols," *IEEE Trans. Mobile Comput.*, vol. 7, no. 1, pp. 50–65, Jan. 2008.

[20] J. S. Pathmasuntharam, A. Das, and A. K. Gupta, "Primary channel assignment based MAC (PCAM)—A multi-channel MAC protocol for multi-hop wireless networks," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, vol. 2, pp. 1110–1115, Mar. 2004.

[21] C. E. Perkins, E. M. B. Royer, and S. Das, "Ad-hoc on-demand distance vector routing," in *Proc. IEEE Workshop Mobile Comput. Syst. Appl. (WMCSA)*, Feb. 1999.

[22] A. Raniwala and T.-C. Chiueh, "Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Mar. 2005.

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

[24] A. Tzamaloukas and J. J. Garcia-Luna-Aceves, "Channel-hopping multiple access," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2000, pp. 415–419.

[25] A. Tzamaloukas and J. J. Garcia-Luna-Aceves, "Channel-hopping multiple access with packet trains for ad hoc networks," in *Proc. IEEE Int. Conf. Mobile Multimedia Commun. (MoMuC)*, Nov. 2000.

[26] S.-L. Wu, C.-Y. Lin, Y.-C. Tseng, and J.-P. Sheu, "A new multi-channel MAC protocol with on-demand channel assignment for multi-hop mobile ad hoc networks," in *Proc. Int. Symp. Parallel Archit. Algorithms Netw. (I-SPAN)*, Dec. 2000, pp. 232–237.

Ting-Yu Lin received the PhD degree in computer science and information engineering (CSIE) from the National Chiao Tung University, Hsinchu, Taiwan. When she graduated, she was recipient of The Phi Tau Phi Scholastic Honor award. From June 2003 to Feburary 2004, she was affiliated with Massachusetts Institute of Technology (MIT) as a research scientist. From March 2004 to August 2005, she worked with Industrial Technology Research Institute of Taiwan as a software engineer. Afterward, she joined the University of Illinois, Urbana-Champaign (UIUC) as a postdoctoral research associate in the period of September 2005 to August 2006 under both the government and university sponsorship. She is currently an associate professor with the Department of Electrical and Computer Engineering, Institute of Communications Engineering, National Chiao Tung University, Hsinchu, Taiwan. Her research interests include wireless communications, mobile computing, WLANs/WPANs, wireless sensor/mesh networking, and green computing. She is a member of ACM.

Kun-Ru Wu received the BS degree in electrical engineering from the National Chung Hsing University, Taichung, Taiwan, in June 2007. He is currently a PhD student with the Institute of Communications Engineering, National Chiao Tung University, Hsinchu, Taiwan. His research interests include wireless mesh networks and linux-based system prototyping.

Guang-Chuen Yin received the BS degree in electronic engineering from National Taipei University of Technology, Taipei, Taiwan, and the MS degree in communications engineering from the National Chiao Tung University, Hsinchu, Taiwan. He is now a firmware engineer at Asus Computer Inc. and focuses on UEFI and ACPI architecture for x86-based mobile platforms. His research interests include multi-channel link- and network-layer protocol designs in wireless environments.