Joint Routing and Spectrum Allocation for Multi-Hop Cognitive Radio Networks with Route Robustness Consideration

Chao-Fang Shih, Wanjiun Liao, and Hsi-Lu Chao

***Abstract*—In this paper, we introduce the concept of "route robustness" for path selection in multi-hop cognitive radio networks. We demonstrate that the aggregate throughput and the robustness of routes determined by the proposed route selection strategy are superior to existing rate-based selection strategies. The rationale behind our approach is to guarantee a basic level of robustness for a set of routes (referred to as skeletons in this paper). Then, we select some routes from this robust route set and determine the spectrum to be allocated on each link along these routes such that the system throughput is maximized. We also design a polynomial time algorithm for this problem, and evaluate our proposed mechanism via simulations. The results show that our proposed algorithm indeed achieves a near optimal solution of this problem for multi-hop overlay CR networks.**

***Index Terms*—Cognitive radio, routing, robustness, spectrum allocation.**

I. Introduction

Wireless spectrum is a scarce resource. However, under the fixed spectrum assignment policy widely used today, most of the spectrum may be underutilized [1]. To solve this problem, Cognitive Radio (CR) [2], [3] was introduced. CR is a technique which allows secondary users (SUs) to access the licensed spectrum when no primary users (PUs) appear on the frequency band (e.g., TV broadcast bands and some cellular bands) or under the condition that the normal operation of PUs will not be interrupted ([4], [5]). With the support of software defined radio (SDR) technology, CR devices can observe and sense the environment, identify spectrum holes, and dynamically adjust its transmission parameters to better utilize the resource, while not interrupting the normal operation of PUs. Accordingly, the channel availability in CR networks is determined by the presence behavior of PUs, and may vary with locations, time and frequency bands. This makes protocol design for CR ad hoc networks much more challenging, as compared with multi-channel ad hoc networks which typically operate under a relatively stable set of available channels.

In this paper, we study the joint routing and spectrum allocation problem in overlay CR ad hoc networks. Each SU can access the network using a channel not currently in used by a PU [4]. Since SUs "temporarily borrow" the spectrum holes from PUs, in order not to affect the normal operations of PUs, SUs must return the spectrum once any of the PUs appears on the channels. As a consequence, the on-going transmissions of SUs on the current path may be disrupted due to link disconnections. To tackle this problem, a typical solution is to perform rerouting and switching to other channels or links currently available. However, rerouting usually introduces extra delay and wastes system resources. Therefore, it is desired to select a "more robust" route that experiences less frequent interruptions.

Finding a robust route in CR ad hoc networks, while important, has not attracted much attention in existing work. Robust routing in traditional networks needs not to consider the impact of PUs, thus they only concentrate on the problem of node mobility and strong sudden interference from neighboring nodes (e.g., [6], [7]). Existing work on routing and channel allocation in CR networks either ignores this issue (e.g., [8]–[14]), or tackles this problem in single flow scenarios (e.g., [15]–[17]). The work in [18] is among the very few which consider this robustness issue in multi-hop multi-flow environments. The solution is rate-biased, i.e., constructing routing metrics based on a combination of many factors including stability. However, in that work, the proper coefficients of each factor are hard to determine, and robustness may not be properly accounted for due to being weighted with other factors. Without ensuring a basic level of robustness, the routing path will become disconnected frequently regardless of how good the other factors are. [19] is another work which exploits the importance of robustness in routing. The mechanism designed in that work is mainly for an underlay CR environment, which is not our focus in this paper.

In this paper, we introduce the concept of "route robustness" for path selection in multi-hop cognitive radio networks. We demonstrate that the aggregate throughput and the robustness

Manuscript received July 13, 2010; revised September 2, 2010; accepted April 25 2011. The associate editor coordinating the review of this paper and approving it for publication was S. Liew.

This work was supported in part by the Excellent Research Projects of National Taiwan University, under Grant Number 97R0062-06, and in part by the National Science Council (NSC), Taiwan, under Grant Number NSC99-2221-E-002-030-MY3.

C.-F. Shih was with the Graduate Institute of Communication Engineering, National Taiwan University, Taipei, Taiwan. She is now with the Electrical and Computer Engineering Department, Georgia Institute of Technology, Atlanta, GA.

W. Liao (corresponding author) is with the Department of Electrical Engineering and the Graduate Institute of Communication Engineering, National Taiwan University, Taipei, Taiwan, and with the Research Center for Information Technology Innovation, Academia Sinica, Taipei, Taiwan (e-mail: wjliao@ntu.edu.tw).

H.-L. Chao is with the Department of Computer Science and Information Engineering, National Chiao-Tung University, Hsin-Chu, Taiwan.

Digital Object Identifier 10.1109/TWC.2011.072011.101249

of routes determined by the proposed route selection strategy are superior to existing rate-based selection strategies. The rationale behind our approach is to guarantee a basic level of robustness for a set of routes. Then, we select some routes from this robust route set and determine the spectrum to be allocated on each link along these routes such that the system throughput is maximized. We also design a polynomial time algorithm for this problem, and evaluate our proposed mechanism via simulations. The results show that our proposed algorithm indeed achieves a near optimal solution of this problem for multi-hop overlay CR networks.

The rest of the paper is organized as follows. In Sec. II, the network model and the problem are described. In Sec. III, the problem is formulated via integer linear programming and a two-stage mechanism is proposed. In Sec. IV, the performance of the proposed mechanism is evaluated via simulations and some design insights are observed. Finally, the paper is concluded in Sec. V.

II. Network Model and Problem Description

A. Network Model and Assumptions

In this paper, we consider an overlay CR model. Nodes exchange information either through a common control channel, as in [13], or in a distributed manner, as in [20]. In our model, the network consists of N CR nodes, L links and C orthogonal frequency bands (channels). F S-D pairs of flows are injected into the network from secondary users (SUs). Each flow is always backlogged, which means that each link, once assigned to a flow, is always busy. We assume that all SU nodes have a common transmission range and a common interference range, as assumed in [13]. Each node is associated with a channel pool which contains the set of channels available for this node. This channel pool may be obtained by sensing the activities of PUs via such methods as feature detection, ambient power sensing, and beacon-based methods [4], and here we assume a perfect sensing result. The condition of each channel may vary with time, location, and radio spectrum, depending on the presence behavior of PUs. As a result, the channel pool associated with adjacent nodes may not be identical. Formally, let $X(i)$ denote the list of available channels for node i. Since the presence behavior of PUs may vary with frequency bands and locations, for nodes i and j, $i \neq j$, $X(i)$ and $X(j)$ may be different. For each link (i, j), $X(i) \cap X(j)$ is called the available channels on link (i, j). Each link together with its associated available channels forms a set of link-channel pairs. For each link, the capacity on different channels may be different.

Suppose that the presence probability of PUs on a channel is accessible from the PHY layer via environment learning. Therefore, it is reasonable to assume that the probability of no PUs on a channel for a period of time is available. To simplify the analysis, we further assume that this probability is independent among all nodes and over all channels. Note that this assumption will affect the way we calculate probability P_{rr} (which will be introduced in Sec. II-B), which will in turn affect the calculation of the "skeleton formation" (which will be introduced in Sec. III-A) when deciding whether the current route satisfies the robustness constraint or not; it will not affect the complexity of the algorithm. If the presence probability of PUs is somehow dependent, the way we calculate the probability P_{rr} (i.e., under this independent assumption) will under-estimate the robustness of this route (i.e., we may skip some eligible routes when selecting routes). Therefore, the route selected by our algorithm will still satisfy the robustness constraint (but the system aggregate throughput will degrade somehow).

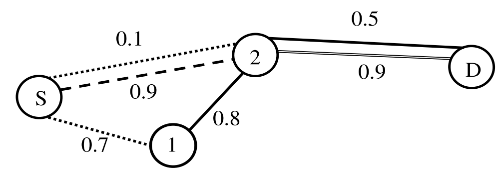


Fig. 1. Example multi-hop CR network topology.

B. The Importance of Route Robustness

The impact of PU disruption on end-to-end connectivity for multi-hop communications is more significant than that in the single hop case. For single hop communications, the source and destination nodes need only consider local channel availability. But for multi-hop communications, each pair of nodes needs to explore the channel availability of the entire route. For example, consider the simple topology shown in Fig. 1, in which link $(1, 2)$ is included in the route for the flow from node S to node D. Suppose that the channel currently in use on link $(1, 2)$ becomes unavailable, and there are no available channels on the consecutive links of link $(1, 2)$ (e.g., link $(S, 1)$). Then, even if there are other channels available on link $(1, 2)$, we may still be unable to reconnect the route by simply switching to other available channels on link $(1, 2)$. As a result, we may still need to find a new route (e.g., the route should go through link $(S, 2)$ because $(S, 1)$ has no available channels) for this flow. In other words, any broken link on a route may incur rerouting and information exchange in order to guarantee the end-to-end connectivity of the entire route. Since rerouting usually introduces extra delay and resource wastage, it is very costly to maintain a route which is prone to link disconnection. Thus, "route robustness" is indeed an important factor for selecting routes in multi-hop cognitive radio networks.

Let $P_s(i, j, c)$ denote the probability that no PU appears on channel c of link (i, j) which is currently in use for transmission. By definition, $P_s(i, j, c)$ can be regarded as the fraction of packets transmitted successfully on channel c of link (i, j) without being interrupted by spectrum mobility (i.e., the presence of PU). For each link (i, j), the larger the value of $P_s(i, j, c)$, the more the number of packets successfully transmitted and the more robust the link-channel pair (i, j, c). Let P_{rr} denote the robustness degree of route r, which is expressed as the product of the values of P_s of the most robust channel on each link along route r, i.e.,

$$P_{rr} = \prod_{\forall (i,j) \in r} \max(P_s(i, j, c) | c \in \{X(i) \cap X(j)\}). \quad (1)$$

Again, we use Fig. 1 to illustrate the concept of P_{rr} by the route going through link $(S, 2)$ and link $(2, D)$. Along this

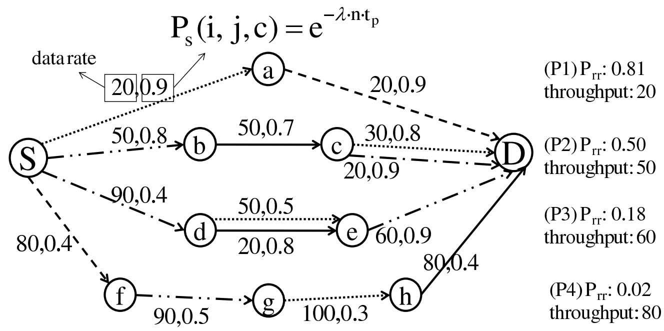


Fig. 2. Example topology with ten CR nodes.

route, there are multiple channels available on each link. The number associated with each channel indicates the value of P_s for the link-channel pair. With this setting, the robustness degree P_{rr} of the route for this S-D pair is expressed by

$$\max\{0.1, 0.9\} \times \max\{0.5, 0.9\} = 0.81.$$

The physical meaning of P_{rr} is explained as follows. Consider the scenario in which there are multiple channels on a link currently in use for an S-D pair. By definition, the likelihood that a PU appears on the most robust channel of the link is the smallest, and a link is connected only if there is at least one channel available. Therefore, it is reasonable to estimate the robustness of a link with the most robust channel of the link. We refer to the set of the most robust channel on each link along a route as the "skeleton" of this route. For example, in Fig. 1, the links with $\{P_s\} = \{0.9, 0.9\}$ form the skeleton of the route going through links $(S, 2)$ and $(2, D)$. Recall that P_{rr} for a route equals the product of the values of P_s of the skeleton for the route. The higher the value of P_{rr}, the more stable (and more robust) this route is. In our problem setting, the value of P_{rr} for each route r should be at least P_m, such that the robustness of each route can be maintained at a certain level. The actual value of P_m is determined by the demands of different flows (and different applications). Since the routes with P_{rr} lower than P_m are too fragile to maintain the connectivity, they will be excluded as candidate routes for packet transmissions in our algorithm. We refer to $P_{rr} \geq P_m$ as the *robustness constraint* for each S-D pair (flow) in the rest of the paper.

Fig. 2 shows a simple example with 10 CR nodes to demonstrate the impact of the robustness constraint on the system performance. The first number on each channel-link pair indicates the rate of the channel-link pair, and the second number indicates the probability $P_s(i, j, c)$. We consider four routes as follows: $(P1) : S \to a \to D$, $(P2) : S \to b \to c \to D$, $(P3) : S \to d \to e \to D$, and $(P4) : S \to f \to g \to h \to D$. The value of P_{rr} and the throughput for the flow on each route are also indicated in Fig. 2 (i.e., the rightmost column in the figure).

We compare four path selection strategies:

- Strategy 1 (robustness-rate rule, with $P_m = 0.8$): we select the route which has the maximum rate, under the robustness constraint with $P_m = 0.8$, i.e., $P_{rr} \geq 0.8$.
- Strategy 2 (robustness-rate rule, with $P_m = 0.5$): we select the route which has the maximum rate, under the robustness constraint with $P_m = 0.5$, i.e., $P_{rr} \geq 0.5$.
- Strategy 3 (effective-rate rule): we select the route with the maximum effective rate of the bottleneck link. The effective rate of a link (i, j) is defined as the summation of the product of the transmission rate of a channel c and its $P_s(i, j, c)$ over all channels on link (i, j), i.e., effective rate of link $(i, j) = \sum_{c \in (X(i) \cap X(j))} r(i, j, c) \times P_s(i, j, c)$, where $r(i, j, c)$ is the Shannon capacity of channel c on link (i, j). The bottleneck link of a route is the link whose effective rate over all links on the route is the minimum. For example, the bottleneck link for $P2$ in Fig. 2 is (b, c) and the effective rate of which is

$$\min\{50 \times 0.8, 50 \times 0.7, 30 \times 0.8 + 20 \times 0.9\} = 35.$$

- Strategy 4 (rate-only rule): we select the route with the maximum rate.

With Strategy 1, $P1$ is the only route to be selected since the constraint is that P_{rr} must exceed 0.8. With Strategy 2, both $P1$ and $P2$ satisfy the robustness constraint, and $P2$ is selected as it has a higher rate. With Strategy 3, the effective rate of the bottleneck link for each route is listed as follows $\{P1 : 20 \times 0.9 = 18$, $P2 : 50 \times 0.7 = 35$, $P3 : 90 \times 0.4 = 36$, and $P4 : 100 \times 0.3 = 30\}$. Thus, $P3$ is selected due to its having the largest bottleneck effective rate among all paths. With Strategy 4, "data rate" is the only factor to consider; thus $P4$ is selected. Note that Strategy 3 and Strategy 4 are rate-based strategies, as in [18].

We then measure the throughput for the S-D pair on each selected route via simulations, based on the four selection strategies described above without performing rerouting and/or rescheduling. In our simulation, the presence behavior of PUs on each channel-link pair follows a Poisson distribution with arrival rate λ. The probability $P_s(i, j, c)$ is then expressed by $P_s(i, j, c) = e^{-\lambda \cdot k \cdot t_p}$, where t_p is the packet transmission time for each link-channel pair (i, j, c), and k is the total number of packets successfully transmitted over the link-channel pair. Here we use $k = 10$ in our simulation for demonstration purpose. Therefore, P_s is the probability that ten packets will be transmitted successfully on each link-channel pair. The simulation results for the four routes in the network shown in Fig. 2 are summarized in Table I (and the results for a random topology can be found in our previous work [21][1]). The "average route lifetime" refers to the average of the maximum time duration that the route remains connected, and the "effective rate" for each route indicates the routing metric used by Strategy 3 (i.e., effective-rate rule). As can be seen, P_{rr} properly reflects the robustness degree for each path. Under this setting, $P1$ (with $P_{rr} = 0.8$) is the most stable path, followed by $P2$ (with $P_{rr} = 0.5$). Both $P3$ and $P4$ are very unstable, due to small values of P_{rr}, i.e., 0.18 and 0.02, respectively. Fig. 3 plots the average throughput for the flow traversing each of the four paths shown in Fig. 2. It depicts that the average throughput for each path decreases with the presence of PUs as time goes by. Although $P3$ and

[1]The results for the random topology are consistent with those shown in Table I. Due to space limitations, we will not include those results in this paper.

TABLE I
AVERAGE ROUTE LIFETIME OF THE PATHS IN FIGURE 2

	$P1$	$P2$	$P3$	$P4$
P_{rr}	0.81	0.5	0.18	0.02
Effective Rate(Strategy 3)	18	35	36	30
Average Route Lifetime	1.57	0.26	0.08	0.02

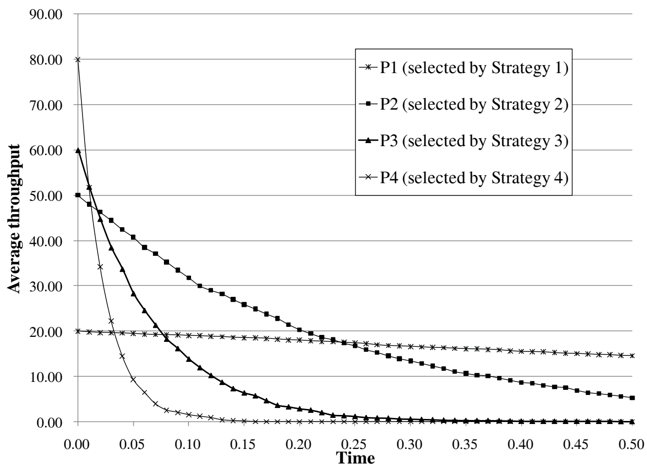


Fig. 3. The throughputs of the paths for the four selection strategies.

$P4$ have higher initial throughputs, their average throughput drops very rapidly due to poor robustness. As far as the aggregate throughput is concerned, $P2$ is superior to $P1$. Both $P3$ and $P4$ are very poor in throughput performance. This is because the "data rate" factor considered in Strategies 3 and 4 has greatly impacted path selection. Ironically, while factoring in "rate" for path selection in both strategies, the resulting aggregate throughputs of both paths are very poor due to choosing unstable paths and resulting in frequent disruption of flow transmissions on the paths. Note that the effective rates of $P2$ and $P3$ are very close (i.e., 35 and 36, respectively). However, their throughput performance is rather different. This is due to the fact that P_{rr} can directly reflect the difference (i.e., 0.50 and 0.18, respectively) in performance while the effective rate of a path cannot. Thus, we learn that ensuring a basic level of robustness degree in path selection is more influential on the path performance.

Based on the observation above, we conclude that the trade-off between robustness and throughput for path selection can be better demonstrated with a proper setting of P_m (i.e., $P1$ (selected by Strategy 1) vs. $P2$ (selected by Strategy 2)), rather than by using a combined factor of data rate and robustness in path selection (i.e., Strategy 3) or even not considering robustness at all (i.e., Strategy 4).

C. Problem Specification

In Sec. II-B, we have demonstrated the importance of route robustness based on an example multi-hop CR networks. The challenge now is how to determine the path to route and the set of channels on each link along the route to use, considering the channel heterogeneity and network dynamics due to the presence of PUs in the networks. In what follows, we attempt to jointly determine routing and spectrum allocation in an overlay CR network such that the selected route is robust (i.e., exceeding a pre-specified value) and the aggregate throughput is maximized.

III. Route Robustness Selection Algorithm

In this section, we propose a joint route selection and spectrum allocation algorithm for multi-hop CR networks. We consider the multi-path scenario for each flow. In our algorithm, routes are determined in two stages. First, the possible skeleton set for each S-D pair in the network is selected so that the robustness constraint can be satisfied (Skeleton Formation). Then, the skeleton to use (which is chosen from the possible skeleton set determined in the first stage) and the channel to allocate on each link along the route skeleton are determined such that the system throughput is maximized (Joint Routing and Spectrum Allocation). Together, the route to go through and the set of channels to use on the links along the route for each flow in the network are determined.

A. Skeleton Formation

We first find all possible route skeletons for each S-D pair, under the constraints that they are loop-free, and that they satisfy the robustness constraint. Since the robustness constraint (i.e., $P_{rr} \geq P_m$) is related to each skeleton, if we want to find an optimal solution, we have to examine all possible skeletons which guarantee the minimum robustness level of each route. However, enumerating all possible skeletons (exponential in number) is computationally expensive. In other words, if we just formulate one optimal problem without pre-selection of routes, the optimal problem will be computationally intractable. Fortunately, the number of skeletons which satisfies the robustness constraint is relatively small. Furthermore, we only need to consider the skeletons which satisfy the robustness constraint, since the route we look for must have a skeleton which falls within the set of skeletons which satisfy the route robustness in order to satisfy the basic robustness level.

We propose a *Skeleton Formation* scheme, similar to *Breadth-First-Search*, to find the skeleton set for each flow in a hop-by-hop manner. For each flow, the skeleton selection starts from the source node of the flow, which is the only end node in the initial skeleton set. For each end node in the set, all its neighboring nodes which satisfy the robustness constraints, together with at least one link-channel pair, are included to the skeleton set. Those skeletons which 1) cannot be extended any further and 2) form a loop will be purged from the skeleton set. This process is repeated until either the destination is reached by all skeletons in the set, or the set is empty. The pseudo code of the detailed algorithm is shown in Table II, and the time complexity of skeleton formation is $O(F \cdot C^{(V-1)} \cdot V^V)$, where V is the number of nodes, F is the number of flows, and C is the number of channels in the network.

For example, suppose we want to find the skeleton set (denoted by Ψ) with $P_m = 0.5$ for the S-D pair in Fig. 1. Initially, $\Psi = \{S\}$. Then, since both neighboring nodes of node S (i.e., nodes 1 and 2), together with at least one

TABLE II
PSEUDO CODE OF SKELETON FORMATION

Skeleton Formation algorithm

Input: Graph G with V nodes and E edges
Sets:$sk\_list, next\_sk\_list.$

for each flow f **do**
  put source node s into sk_list
  repeat
    for each route r in sk_list **do**
      $v \leftarrow$ the end node of r
      if $v ==$ destination node d **then**
        $next\_sk\_list \leftarrow r$
      else
        remove r from sk_list
        for each neighbor u of v, each available channel c **do**
          make a new route $r' = merge(r, v-u-c)$
          if r' has no loop and $P_{rr}(r') \geq P_m$ **then**
            $next\_sk\_list \leftarrow r'$
          else
            delete r'
          end if
        end for
      end if
    end for
    swap $next\_sk\_list$ and sk_list /*sk_list is empty now*/
  until (sk_list is empty) or (the destination d is reached by all routes in sk_list)
  store sk_list for flow f
end for

channel-link pair, satisfy the robustness constraint, the set Ψ is then updated as follows: $\Psi = \{S \rightarrow 2(0.9), S \rightarrow 1(0.7)\}$. Note that skeleton is not included due to its violating the robustness constraint. We next consider the two end nodes in Ψ, i.e., nodes 1 and 2. Since $S \rightarrow 2(0.9) \rightarrow 1(0.8)$, $S \rightarrow 2(0.9) \rightarrow D(0.9)$, and $S \rightarrow 1(0.7) \rightarrow 2(0.8)$, all satisfy the robustness constraint, and among them, $S \rightarrow 2(0.9) \rightarrow D(0.9)$ has reached the destination, the set is then updated as $\Psi = \{S \rightarrow 2(0.9) \rightarrow D(0.9), S \rightarrow 2(0.9) \rightarrow 1(0.8), S \rightarrow 1(0.7) \rightarrow 2(0.8)\}$. Next, we extend the two end nodes in the set Ψ. This time, $S \rightarrow 1(0.7) \rightarrow 2(0.8) \rightarrow D(0.9)$ has reached the destination. The other two skeletons, i.e., $S \rightarrow 1(0.7) \rightarrow 2(0.8) \rightarrow 1(0.8)$ and $S \rightarrow 2(0.9) \rightarrow 1(0.8) \rightarrow 2(0.8)$, while satisfying the robustness constraint, result in loops. Therefore, they are purged from the set. Since all skeletons in the set have reached the destination, the process stops and the final skeleton set is given by:

$$\begin{aligned}\Psi = \{&S\rightarrow 2(0.9) \rightarrow D(0.9),\\ &S\rightarrow 1(0.7) \rightarrow 2(0.8) \rightarrow D(0.9)\}.\end{aligned}$$

B. Truncated Skeleton Formation

As shown in the previous section, the time complexity of the skeleton formation is exponential. To reduce the time complexity, we next propose a polynomial time algorithm called *Truncated Skeleton Formation* for skeleton formation. Our approach is that we find all possible skeletons of routes for each S-D pair under the constraints that they are loop-free, the robustness constraint is satisfied, and the hop count is smaller than a predefined number MAX_HOP. Finding all skeletons whose hop counts are within a pre-determined MAX_HOP can be done in polynomial time. Moreover, since longer routes (i.e., with a larger hop count) have a lower probability to be chosen (the selection process will be described later), the performance degradation will be acceptable if we can choose a reasonable value of MAX_HOP.

Obviously, the value of MAX_HOP greatly affects the performance of *Truncated Skeleton Formation*, which is $O(F \cdot C^{MAX\_HOP} \cdot V^{MAX\_HOP})$. A larger value of MAX_HOP results in higher time complexity and less performance degradation. Thus, we would like to select the value of MAX_HOP that is not too high so that the time complexity is acceptable, and is not too low so that the performance will only be degraded slightly. Actually, the skeleton that will be chosen by the proposed *Joint Routing and Spectrum Allocation* algorithm (which will be described in Sec. III-C) usually has a hop count which is just one or two hops more than the minimum hop count or may even equal the minimum hop count. This is because the skeleton with a large hop count can hardly (i.e., with a very small probability) satisfy the robustness requirement. Moreover, skeletons with a large hop count also have a higher probability to introduce more interference to other routes, leading to lower overall throughput. Therefore, it is desired to choose routes with smaller hop counts. To speed up the skeleton formulation, we want to set a limit on the hop count to ensure polynomial time complexity while imposing minor impact on the performance. In what follows, we propose a reasonable, but loose, bound for the hop count. We will shortly show in the simulation that the performance with such a setting is very close to the optimal.

The value of MAX_HOP for a flow can be determined by considering the maximum complexity a system can handle. Alternatively, we can set the value of MAX_HOP as the maximum possible hop count for a route that satisfies the robustness constraint, i.e., $P_m \leq \alpha^{MAX\_HOP}$. By taking a logarithm operation on both sides, the value of MAX_HOP can be determined as follows.

$$MAX\_HOP = \lceil \log_\alpha P_m \rceil, \tag{2}$$

where α represents the estimated value of P_s for each link-channel pair. α is a positive value smaller than one and can be expressed in (3), accounting for both the long term and short term effects of Ps:

$$\alpha = E(P_s) + \beta Std(P_s), \tag{3}$$

where $E(P_s)$ and $Std(P_s)$ are the expected value and the standard deviation of Ps, respectively, and β is a tunable value. A large value of β means the result favors the short term variation, while a small value of β gives more focus on the long term behavior. The long term effect is captured by $E(P_s)$, and the short term network dynamic can be described by some deviation of $E(P_s)$, i.e., $Std(P_s)$. Note that the values of both $E(P_s)$ and $Std(P_s)$ can be obtained by a weighted moving average.

C. Joint Routing and Spectrum Allocation Algorithm

Having selected the skeleton set for an S-D pair in the first stage, we next determine which skeletons and which set of channels to use on each link along the determined skeletons

TABLE III
THE ILP FORMULATION FOR JOINT ROUTING AND SPECTRUM ALLOCATION

Integer Linear Programming Formulation

Variables: $b_c(i,j), u(f,k), a(f,k)$.
Constants: $r_{max}(i,j,c), R_{max}$.
Objective function: $\max \sum_{r(f,k)} a(f,k)$.
Constraints:
Integer constraints:

The value of $b_c(i,j) = \{0,1\}$; The value of $u(f,k) = \{0,1\}$

Interference constraints:

$$\sum_{i \in T(j)} b_c(i,j) + \sum_{k \in T(j)} b_c(i,j) \leq 1, \forall j \in N, c \in C$$

$$b_c(i,j) + \sum_{l \in T(k)} b_c(k,l) \leq 1, \forall k \in I(j), k \neq i, k \neq j, \text{and } \forall b_c(i,j)$$

Link capacity constraints:

$$\sum_{r(f,k) \in \Re(i,j)} a(f,k) \leq \sum_{c \in (X(i) \cap X(j))} r_{max}(i,j,c) \times b_c i,j, \forall (i,j)$$

Skeleton constraints: (Robustness consideration)

$$a(f,k) \leq u(f,k) \times R_{max}, \forall r(f,k)$$
$$b_c(i,j) \geq u(f,k), \forall (i,j,c) \in r(f,k)\text{'s skeleton}$$

for routing. The joint routing and channel assignment problem for multi-hop wireless network is NP-complete, as shown in [13]. To solve this problem, we formulate it via an Integer Linear Programming (ILP) model, the objective of which is to maximize the system throughput (i.e., aggregate throughput) in the network. Let $r(f,k)$ denote route k for flow f, and $a(f,k)$ denote the data rate of route $r(f,k)$. Mathematically, the objective function is to maximize the summation of $a(f,k)$ for all flows f, i.e.,

$$\text{maximize} \sum_{r(f,k)} a(f,k).$$

Our formulation takes into account the channel heterogeneity and network dynamics of CR networks (due to the presence behavior of PUs). The detailed formulation is summarized in Table III. The set of constraints for this problem is explained as follows.

1) Interference Constraint: Let $b_c(i,j)$ denote the binary indicator for the assignment of channel c on link (i,j). If channel c is assigned to link (i,j), $b_c(i,j) = 1$; otherwise, $b_c(i,j) = 0$. There are two types of interference constraints that $b_c(i,j)$ must meet.

- Each node cannot receive data from more than one node on the same channel, and each node cannot transmit and receive data simultaneously on the same channel. Mathematically,

$$\sum_{i \in T(j)} b_c(i,j) + \sum_{k \in T(j)} b_c(j,k) \leq 1, \quad \forall j \in N, c \in C, \quad (4)$$

where $T(j)$ denotes the set of nodes within the transmission range of node j.

- When a node receives data from another node on a channel, the other nodes which are located within the interference range of this receiving node cannot transmit on the same channel, i.e.,

$$b_c(i,j) + \sum_{l \in T(k)} b_c(k,l) \leq 1, \quad \forall k \in I(j), k \neq i, k \neq j, \text{and } \forall b_c(i,j), \quad (5)$$

where $I(j)$ denotes the set of nodes within the interference range of node j.

Note that we consider only one node k in $I(j)$ each time in (5). Therefore, for any two (or more) nodes, say, k_1 and k_2, which are both located within $I(j)$, they are allowed to transmit simultaneously if the following two conditions hold: i) $b_c(i,j)$ is equal to 0, and ii) k_1 and k_2 will not interfere with each other. In addition, since we restrict the value of $b_c(i,j)$ to be an integer, if there are three (or more) links which interfere with one another, resulting in a clique in a conflict graph, there is only one node that is able to transmit. Thus, the solution space defined by these interference constraints is always feasible.

2) Skeleton Constraint (i.e., Robustness Consideration): Based on the skeleton set selected by the skeleton formation process, we can find many routes for each flow. Now, we want to determine the route to be used. Let $u(f,k)$ denote the binary indicator for route $r(f,k)$. If we choose route $r(f,k)$ to transmit data, $u(f,k) = 1$; $u(f,k) = 0$, otherwise. Recall that $a(f,k)$ denotes the data rate of route $r(f,k)$. If we do not use the skeleton of $r(f,k)$, $a(f,k)$ equals zero, and if any of the link-channel pair (i,j,c) on route $r(f,k)$ is not assigned, we cannot use the skeleton of this route. Therefore, this leads to the following two flow constraints:

$$a(f,k) \leq u(f,k) \times R_{max}, \forall r(f,k), \quad (6)$$
$$b_c(i,j) \geq u(f,k), \forall (i,j,c) \in r(f,k)\text{'s skeleton}, \quad (7)$$

where R_{max} is a large number used to set up the constraint (i.e., to ensure $a(f,k)$ becomes zero if $u(f,k)$ is zero), and it can be set as the maximum possible throughput that a route can achieve.

3) Link Capacity Constraint: The total data rates over a link must not exceed the maximum data rate of all assigned channels on that link, i.e.,

$$\sum_{(f,k) \in \Re(i,j)} a(f,k) \leq \sum_{c \in X(i) \cap X(j)} r_{max}(i,j,c) \times b_c(i,j), \quad \forall (i,j), \quad (8)$$

where $\Re(i,j)$ is the set of routes which pass through link (i,j), and $r_{max}(i,j,c)$ is the maximum throughput of channel c on link (i,j).

Note that since we consider an overlay CR network for spectrum sharing, SUs and PUs will not co-exist in the network (otherwise, SUs will interfere with PUs in transmissions). In addition, our interference constraints guarantee that the transmissions among SUs will not interfere with one another. Thus, each flow can use the full capacity of the channels assigned to it.

D. Joint Routing and Spectrum Allocation with Polynomial Time Complexity

The joint routing and spectrum allocation is an NP-complete problem [13]. To reduce the time complexity, here we propose a polynomial time algorithm for joint routing and spectrum allocation. Like the optimal joint routing and spectrum allocation in Sec. III-C, we determine which skeletons and which set of channels to use on each link along the determined skeletons for routing in our heuristic. Here, instead of solving the Integer Linear Programming problem described in Sec. III-C, we relax the integer constraints of $b_c(i,j)$ and $u(f,k)$, and solve the relaxed Linear Programming problem iteratively. Since the iterative Linear Programming is shown to be near optimal in [13], it follows that our algorithm is a near-optimal solution with polynomial time complexity.

1) Iterative Linear Programming: The iterative Linear Programming algorithm works as follows. Initially, for all channels c on link (i,j) and all skeletons (i.e., the resulting skeleton sets from *skeleton formation*), $b_c(i,j)$ and $u(f,k)$ are marked as undetermined. In each iteration, Linear Programming is first applied; then, in the result of Linear Programming, among all the undetermined $u(f,k)$, the $u(f,k)$ whose value is one is marked as determined; among all the undetermined $u(f,k)$ whose value is less than 1, the $u(f_1,k_1)$ with the largest corresponding throughput $a(f_1,k_1)$ is selected. If the value of the selected $u(f_1,k_1)$ is 0, that means all the values of $u(f,k)$ are either 1 or 0; otherwise, we set the value of $u(f_1,k_1)$ to 1 and mark $u(f_1,k_1)$ as determined. By setting the value of $u(f_1,k_1)$ to 1, according to all the constraints in Sec. III-B, we will have to set the values of other $b_c(i,j)$ and $u(f,k)$ to 1 or 0. We assign all these values, mark them as determined, and then apply Linear Programming again with the new values of $b_c(i,j)$ and $u(f,k)$ serving as inputs. After all $u(f,k)$ have been set to 1 or 0, we repeat the same procedure to $b_c(i,j)$. When all $b_c(i,j)$ and $u(f,k)$ become 1 or 0, the algorithm stops. The pseudo code of the iterative Linear Programming is summarized in Table IV.

TABLE IV
PSEUDO CODE OF POLYNOMIAL TIME JOINT ROUTING AND SPECTRUM ALLOCATION

Iterative Linear Programming

```
Variables: b_c(i,j), u(f,k)
 1: while there is u(f,k) whose value is not determined do
 2:     Apply the Linear Programming
 3:     max = 0, max_id = (0,0)
 4:     for all the undetermined u(f,k) do
 5:         if u(f,k) == 1 then
 6:             set u(f,k) to 1, mark u(f,k) as determined
 7:         else if a(f,k) > max then
 8:             max = a(f,k), max_id = (f,k)
 9:         end if
10:     end for
11:     if max == 0 then break
12:     else
13:         set u(max_id) to 1, mark u(max_id) as determined
14:         determine other b_c(i,j) and u(f,k) according all the constraints
15:     end if
16: end while
17: repeat
18:     line 1 to 16 for b_c(i,j) and select the largest b_c(i,j) each time
19: until all b_c(i,j) are determined
```

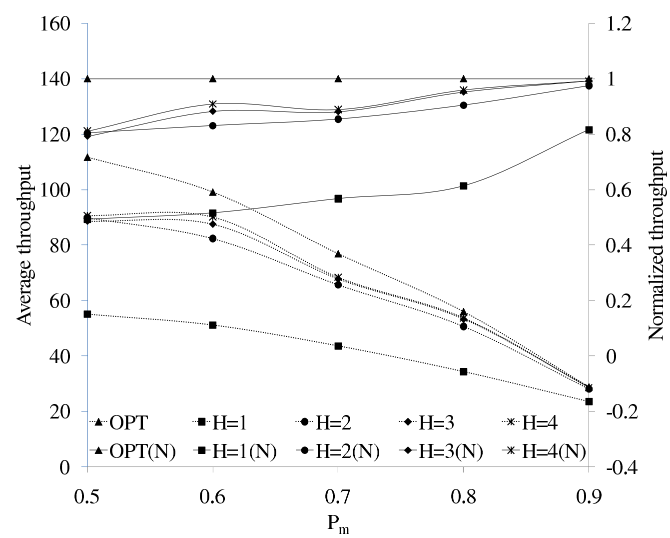


Fig. 4. Comparison of optimal and heuristic results.

IV. PERFORMANCE EVALUATION

In this section, we evaluate the performance of our algorithm via simulations based on an optimization tool called LINGO together with a C++ simulator. In each simulation, the location of the nodes, the source and destination pairs, and the availability channel pool of each node are randomly determined. Each setting is run on 100 randomly generated topologies.

A. Comparison of Optimal Result and Polynomial Time Complexity Result

We compare our polynomial time complexity result with the optimal result based on a 50×50, 20-node topology as shown in Fig. 4. The optimal result is obtained with the "skeleton formulation" and the "joint routing and spectrum allocation algorithm," while the polynomial time complexity result is obtained with the "truncated skeleton formation" and the "joint routing and spectrum allocation with polynomial time complexity." The value of P_s is randomly selected over [0.50, 0.99]. Fig. 4 shows the average system throughput. As can be seen, the higher the P_m, the lower the average system throughput. This is because when P_m increases, the number of skeletons (routes) that can satisfy P_m is reduced, demonstrating the trade-off between throughput and robustness.

We next observe the impact of the value of MAX_HOP on the performance. In Fig. 4, the larger the value of MAX_HOP selected, the better the throughput performance. However, no matter how large the MAX_HOP we choose, there is always a "performance gap" between the heuristic results and the optimization results. This "performance gap" is introduced by the iterative Linear Program, and there is no way to overcome this problem even if we select a larger MAX_HOP. Fig. 4 also shows the normalized results (i.e., the curves denoted "(N)") with respect to the optimal ones. With an appropriate value of MAX_HOP ($H \geq 4$), the normalized ratio can exceed 0.8. It can also be seen that the

TABLE V
RELATIONSHIP BETWEEN THROUGHPUT AND TIME COMPLEXITY

Topology with a larger hop count		
Hop count	Average throughput	Time complexity level
3	0.00	$\sim 10^3$
4	0.30	$\sim 10^4$
5	1.76	$\sim 10^5$
6	1.95	$\sim 10^6$

Topology with a smaller hop count		
Hop count	Average throughput	Time complexity level
1	37.37	$\sim 10^1$
2	73.51	$\sim 10^2$
3	84.70	$\sim 10^3$
4	85.83	$\sim 10^4$

way we estimate MAX_HOP leads to a reasonable upper bound. For example, the value of MAX_HOP given by (2) for $P_m = \{0.5, 0.6, 0.7, 0.8, 0.9\}$ is $\{5, 5, 5, 4, 4\}$. Note that with $H = 5$, $H = 6$, and some other larger values, which are not shown in the paper due to space limitations, the results are almost the same as that with $H = 4$ because routes with hop counts larger than 4 are seldom chosen under our topology setting. Thus, $\{5, 5, 5, 4, 4\}$ is a reasonable setting for MAX_HOP in this case.

B. Balance Between Performance and Time Complexity

In the previous discussion, we consider the optimal case in which the activities of all PUs will not change during the process of finding all possible routes. Thus, a larger value of MAX_HOP results in better performance. This may not be the case if the selection process lasts much longer. While the optimal algorithm serves as a performance upper bound so we can understand how much we sacrifice in performance in order to gain lower time complexity, the proposed polynomial time algorithm is able to reduce the processing time and therefore is a more practical solution. Here, we further propose a way to select a value of MAX_HOP which strikes a good balance between the performance and time complexity.

1) The Selection of MAX_HOP: Since no matter how large the value of MAX_HOP is chosen, there is always a "performance gap" between the polynomial time results and the optimal results. Moreover, the value of MAX_HOP greatly affects the time complexity of our polynomial time complexity algorithm. The selection of MAX_HOP greatly affects the tradeoff of performance and time complexity.

Selecting routes with a smaller hop count enjoys many benefits. Since each hop in a route may become the bottleneck of the throughput, the route with a smaller hop count will have i) a higher probability to achieve larger throughput, ii) a higher probability to satisfy P_m, and iii) a lower probability to generate interference to other routes. However, small-hop-count routes have a smaller probability to reach the destination. Thus, here we introduce $possible\_min\_hop$, which is the minimal hop count among all routes which are loop-free, have no self-interference, and satisfy P_m. Although $possible\_min\_hop$ may be a good choice, we still need to have an upper bound on the hop count to avoid the case that the value of $possible\_min\_hop$ is too large. It is counter-productive to select a very large hop count since the throughput improvement that can be achieved is indeed very minor. For example, Table V shows the relationship between the throughput performance and time complexity level of two different topologies: one with a larger hop count, and the other with a smaller hop count. As shown in Table V, it will be good to select the hop count with which the throughput starts to converge (e.g., from hop count=1 to hop count=2, the throughput can be improved by about 36 units with low time complexity) in the topology with a smaller hop count. However, in the topology with a larger hop count, it will only be a waste of time to select the hop count with which the throughput starts to converge since the throughput improvement is very small but the time complexity is very high (e.g., from hop count=4 to hop count=5, the throughput can only be improved by about 1.5 units with high time complexity). Here, we simply use the result described is Sec. III-B-(1) as our time complexity bound. Based on the discussion above, the selection rule is given by:

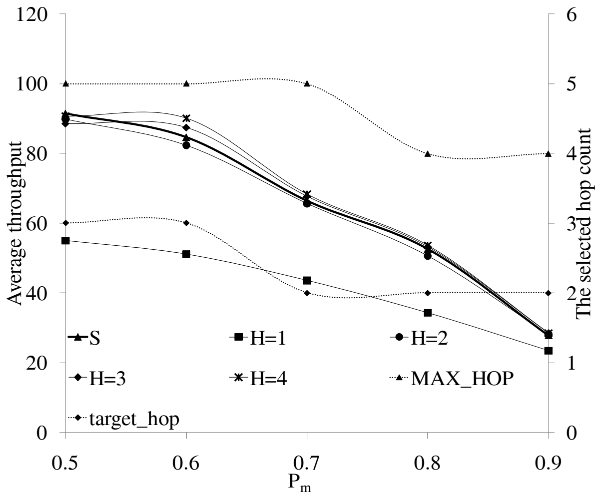
$$target\_hop = \min(possible\_min\_hop, \frac{\log P_m}{\log \alpha}). \quad (9)$$

By substituting MAX_HOP with $target\_hop$ in the *Truncated Skeleton Formation* algorithm, we can obtain the result. Note that the value of $possible\_min\_hop$ automatically reflects the three factors which may affect the selection of maximum hop count mentioned in Sec. IV-C-(1), and the time complexity bound we used will be affected by the value of P_m and the distribution of P_s.

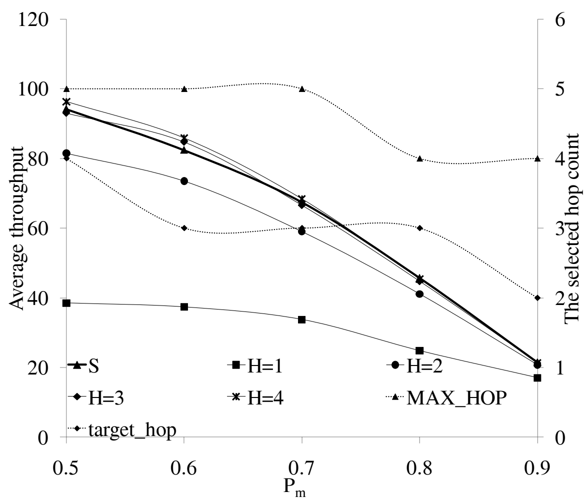
2) Simulation Results: The simulation results in Fig. 5 show that our selection rule works well. The topology used in Fig. 5(a) is a 50×50 network with 20 nodes, and in Fig. 5(b), a 100×100 network with 80 nodes. Fig. 5(c) is a special topology where we separate the source nodes and the destination nodes apart as much as possible. The average throughputs of our selection rule, which is denoted by "S" in the figures, always approach the best performance among all. In Figs. 5(a) and 5(b), the selection rule adaptively chooses the route with a smaller hop count when P_m grows larger, and chooses a larger hop count when the topology grows larger. In Fig. 5(c), the throughput improvement is very small and becomes negligible when the selected hop is equal to the time complexity bound (MAX_HOP). This also demonstrates that the time complexity bound is reasonable.

V. CONCLUSION

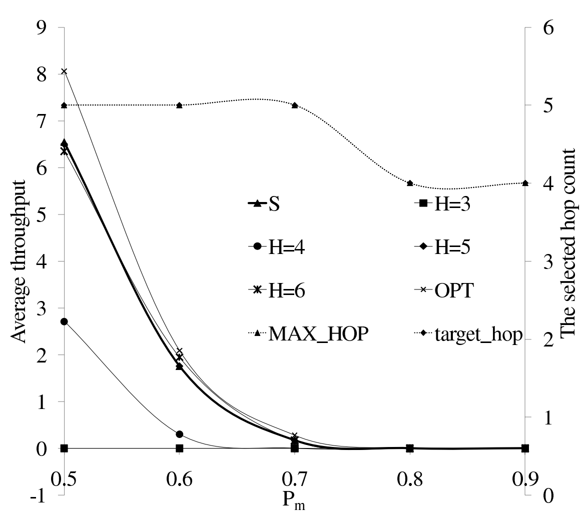
In this paper, we study the joint routing and spectrum allocation problem in multi-hop cognitive radio networks. We take into account the channel heterogeneity property and the channel dynamics of CR networks. We demonstrate that route robustness greatly impacts system performance. We then propose an optimal solution to jointly determining which routes to use and how to allocate spectrum on each link along the routes such that the system throughput is maximized. The rationale behind our solution is to guarantee a basic level of robustness for a set of routes, based on which routes are selected and the channel on each link along the routes is allocated. We also propose an algorithm with polynomial time



(a) For a network with 50 × 50, 20 nodes



(b) For a network with 100 × 100, 80 nodes



(c) For a special topology

Fig. 5. Simulation results for average throughput and selected hop count of selection rule.

complexity. The performance of the proposed mechanism is evaluated via simulations. The results show that the solution obtained by the polynomial time complexity algorithm is near optimal and can achieve a good balance between performance and time complexity.

REFERENCES

[1] FCC, Et docket no 03-222 notice of proposed rule making and order, Dec 2003.

[2] S. Haykin, "Cognitive radio: brain-empowered wireless communications," *IEEE J. Sel. Areas Commun.*, 2005.

[3] R. W. Thomas, L. A. DaSilva, and A. B. MacKenzie, "Cognitive networks," *IEEE DySPAN*, 2005.

[4] I. F. Akyildiz, W. Y. Lee, M. C. Vuran, and S. Mohanty, "Next generation/dynamic spectrum access/cognitive radio wireless networks: a survey," *Computer Networks*, Sep. 2006.

[5] D. Cabric, S. M. Mishra, D. Willkomm, R. W. Broderson, and A. Wolisz, "A cognitive radio approach for usage of virtual unlicensed spectrum," *IST Mobile Wireless Comm. Summit*, 2005.

[6] M. Grossglauser and M. Vetterli, "Locating nodes with ease: last encounter routing in ad hoc networks through mobility diffusion," *IEEE INFOCOM*, 2003.

[7] G. Parissidis, M. Karaliopoulos, T. Spyropoulos, and B. Plattner, "Interference-aware routing in wireless multihop networks," *IEEE Trans. Mobile Comput.*, 2010.

[8] C. Peng, H. Zheng, and B. Y. Zhao, "Utilization and fairness in spectrum assignment for opportunistic spectrum access," *ACM Mobile Networks and Applications*, 2006.

[9] H. Zheng and C. Peng, "Collaboration and fairness in opportunistic spectrum access," *IEEE ICC*, 2005.

[10] H. Zheng and L. Cao, "Device-centric spectrum management," *IEEE DySPAN*, 2005.

[11] M. Thoppian, S. Venkatesan, R. Prakash, and R. Chandrasekaran, "Mac-layer scheduling in cognitive radio based multi-hop wireless networks," *IEEE WoWMoM*, 2006.

[12] L. Cao and H. Zheng, "Distributed spectrum allocation via local bargaining," *IEEE SECON*, 2005.

[13] Y. T. Hou, Y. Shi, and H. D. Sherali, "Optimal spectrum sharing for multi-hop software defined radio networks," *IEEE INFOCOM*, 2007.

[14] G. Cheng, W. liu, Y. Li, and W. Cheng, "Joint on-demand routing and spectrum assignment in cognitive radio networks," *IEEE ICC*, 2007.

[15] H. Kushwaha, Y. Xing, R. Chandramouli, and H. Heffes, "Reliable multimedia transmission over cognitive radio networks using fountain codes," *Proc. IEEE*, 2008.

[16] D. Willkomm, J. Gross, and A. Wolisz, "Reliable link maintenance in cognitive radio systems," *IEEE DYSPAN*, 2005.

[17] T. Fujii and Y. Yamao, "Multi-band routing for ad-hoc cognitive radio networks," *IEEE SDR*, 2006.

[18] M. Sharma, A. Sahoo, and K. D. Nayak, "Channel selection under interference temperature model in multi-hop cognitive mesh networks," *IEEE DySPAN*, 2007.

[19] H. Khalife, S. Ahuja, N. Malouch, and M. Krunz, "Probabilistic path selection in opportunistic cognitive radio networks," *IEEE Globecom*, 2008.

[20] Y. R. Kondareddy, P. Agrawal, and K. Sivalingam, "Cognitive radio network setup without a common control channel," *IEEE MILCOM*, 2008.

[21] C.-F. Shih and W. Liao, "Exploiting route robustness in joint routing and spectrum allocation in multi-hop cognitive radio networks," *IEEE WCNC*, 2010.

Chao-Fang Shih received the B.S. degree in Electronic Engineering and the M.S. degree in Communication Engineering from the National Taiwan University, Taiwan, in 2006 and 2008 respectively. She is now a Ph.D. student in Georgia Institution of Technology, Atlanta, Georgia.

Wanjiun Liao received her Ph.D. degree in Electrical Engineering from the University of Southern California, USA, in 1997. She is Professor of Electrical Engineering, National Taiwan University (NTU), Taipei, Taiwan, and an Adjunct Research Fellow of Research Center for Information Technology Innovation, Academia Sinica, Taiwan. Her research focuses on the design and analysis of wireless and multimedia networking protocols. Prof. Liao is currently an Associate Editor of IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, and she has served on the editorial board of IEEE TRANSACTIONS ON MULTIMEDIA. She has also served on the organizing committees of many international conferences, including serving as Tutorial Co-Chair of IEEE INFOCOM 2004, TPC Area Chair of IEEE International Conferences on Multimedia and Expo (ICME) 2004, TPC Vice Chair of IEEE GLOBECOM 2005 Symposium on Autonomous Networks, TPC Co-Chair of IEEE GLOBECOM 2007 General Symposium, TPC Co-Chair of IEEE VTC 2010 Spring and TPC Co-Chair of IEEE ICC 2010 Next Generation Networking and Internet Symposium. Prof. Liao has received many research awards and recognitions from different government and professional organizations, including IEEE and ACM. She was a recipient of the Republic of China (R.O.C.) Distinguished Women Medal in 2000 and was elected as a Distinguished Lecturer of IEEE Communications Society (2011-2012). She is a Fellow of IEEE.

Hsi-Lu Chao received the B.S. degree in Electronic Engineering from Fengchia University, Taiwan, in 1992, the M.S. degree in Electrical Engineering from the University of Southern California, Los Angeles, in 1996, and the Ph.D. degree in Electrical Engineering from the National Taiwan University, Taiwan, in 2004. Since August 2004, she has been an Assistant Professor, Department of Computer and Information Science, National Chiao Tung University. Her researches cover cognitive radio networks, mmWAVE personal area networks, 4G networks, cooperative communications, and heterogeneous wireless networks.