Efficient Recovery Algorithms for Wireless Mesh Networks with Cognitive Radios

Roberto Hincapie, Li Zhang, Jian Tang, Guoliang Xue, Richard S. Wolff and Roberto Bustamante

***Abstract*—Cognitive radios allow unlicensed wireless users to access channels that are in the licensed spectrum bands. However, in a wireless network with cognitive radios, when a licensed user becomes active on a channel in a certain area, nodes and links that were using that channel must release it, which will cause traffic failures. Simple and effective recovery schemes are needed to re-allocate available resources for the failed traffic. In this paper, we study the failure recovery in wireless mesh networks with cognitive radios. We formally formulate the corresponding problems as integer linear programming problems. By solving them, we can obtain optimal solutions. Moreover, an efficient distributed heuristic algorithm is presented for fast recovery. Simulation results show that the performance given by our distributed algorithm is close to that of the optimal solutions.**

***Index Terms*—Cognitive radio, wireless mesh network, recovery, mathematical programming.**

I. Introduction

Wireless Mesh Networks (WMNs) are envisioned to provide various data and multimedia applications, such as Internet access, neighborhood gaming, Video-on-Demand (VoD) and emergency communications, to wireless users in the future [1]. These applications usually involve a large volume of traffic, which needs a large amount of bandwidth for packet delivery.

An unlicensed wireless user with a cognitive radio (a.k.a secondary user) can sense and access an under-utilized spectrum band opportunistically even if it is licensed, as long as no licensed wireless users (a.k.a primary users) are active on that band, which will lead to better resource utilization therefore larger bandwidth and higher throughput. Hence, cognitive radios can enhance the functionalities of a WMN system and help it better serve its applications. However, the spectrum (channel) availability fluctuates over time and location. In a WMN with cognitive radios, when a primary user becomes active on a channel in a certain area, nodes and links that were using that channel must release it, which will cause traffic failures. Simple and effective recovery schemes are needed to re-allocate the available resources for the failed traffic.

In this paper, we study failure recovery in WMNs with cognitive radios. With consideration for the simplicity and control overhead, only *locally* available resources are allowed to be used to fix the failed traffic on each link. Specifically, we consider two recovery schemes: *single-hop* and *two-hop*. The two-hop recovery scheme fixes the failed traffic on a link by re-allocating resources available on that link, or by re-routing it via a common neighbor of two end nodes and allocating resources accordingly. However, the single-hop recovery scheme does not allow re-routing. The recovery problem is a hard problem in a cognitive radio system since different channels can support quite different transmission ranges and data rates. Therefore, we first formulate the corresponding problems as Integer Linear Programming (ILP) problems. By solving them, optimal solutions can be obtained. Moreover, an efficient distributed algorithm is presented for fast recovery, whose performance is justified by NS2-based simulations. To our best knowledge, This is the first work addressing the recovery from failures caused by channel availability changes in cognitive radio based WMNs.

The rest of the paper is organized as follows. We discuss related work in Section II. The system model and the ILP formulation are described in Section III. The distributed algorithm is presented in Section IV. We present simulation results in Section V and conclude the paper in Section VI.

II. Related Work

Research on networking with cognitive radios is still in its infancy [2]. The spectrum allocation, scheduling and routing in multi-hop wireless networks with cognitive radios have been studied by recent works. The authors of [13] introduced the concept of time-spectrum block and proposed the algorithms to allocate such blocks to meet certain performance goals. In [14], Zheng *et al.* developed a graph-theoretic model to characterize the spectrum access problem and devised a set of heuristic algorithms to find high throughput and fair solutions. In [12], a novel layered graph was introduced to model spectrum access opportunities, which was then used to develop joint spectrum allocation and routing algorithms. A mixed integer non-linear programming based algorithm was presented to solve a joint spectrum allocation, scheduling and routing problem in [6]. In addition, the authors of [11] presented distributed algorithms for joint spectrum allocation, power control, routing, and congestion control. In a very recent work [5], we presented a distributed routing protocol which can select a route and allocate resources for a connection request to satisfy its end-to-end bandwidth requirement.

The channel assignment has also been studied in [8], [9] for traditional WMNs with homogeneous channels which are always available for every mesh node. This work focuses on failure recovery for WMNs with cognitive radios with consideration for dynamic channel availability, channel heterogeneity

This research is funded in part by NSF grants CNS-0721880, CNS-0721803 and CNS-0519403.

Roberto Hincapie is with the Department of Telecommunications Engineering, Universidad Pontificia Bolivariana, Medellin, Colombia. Email: roberto.hincapie@upb.edu.co. This work was done when he visited Montana State University. Li Zhang and Jian Tang are with the Department of Computer Science at Montana State University, Bozeman, MT 59717-3880. Email: {zhang,tang}@cs.montana.edu. Guoliang Xue is with the Department of Computer Science and Engineering at Arizona State University, Tempe, AZ 85287-8809. Email: xue@asu.edu. Richard S. Wolff is with the Department of Electrical and Computer Engineering at Montana State University, Bozeman, MT 59717-3780. Email: rwolff@montana.edu. Roberto Bustamante is with the Department of Electric and Electronic Engineering at Universidad de los Andes, Bogota, Colombia. Email: rbustama@uniandes.edu.co.

and the end-to-end bandwidth constraint, which has not been carefully addressed by any of the previous works.

III. Problem Formulation

We consider a WMN consisting of stationary mesh nodes with locations known, each of which is equipped with a cognitive radio. The available spectrum is divided into a set of non-overlapping channels. During a certain period, a set of channels are available to a particular mesh node. The channel availability may change over the time due to the appearance of primary users in the same area [2]. Any proposed spectrum sensing scheme can be used to detect the locally available channels [2]. We also assume that each node has a low-cost control radio (no need to be a cognitive radio), which is used for exchanging control messages over a common control channel. In addition, the TDMA scheme is used at the MAC layer, which divides the time domain into timeslots with fixed durations and groups them into frames of T timeslots each. A transmission block i is defined as a timeslot-channel pair (t_i, h_i) with timeslot t_i and channel h_i, whose capacity $c_i = \frac{C_{h_i}}{T}$ (i.e., the data rate that can be supported by this block), where C_{h_i} is the capacity of channel h_i. There are a set of end-to-end flows in the network. For each flow, a single source to destination path is assumed to be established for routing. In a link, multiple transmission blocks usually need to be allocated to carry traffic for flows going through it. Any routing and spectrum allocation algorithms such as the algorithms presented in [5] can be used to select route and determine transmission block allocation in advance.

Every transmitter is assumed to transmit at a fixed transmission power level. Hence, there are a fixed transmission range R_h and a fixed interference range I_h (typically 2 to 3 times the transmission range [8]) associated with channel h. Note that the transmission and interference ranges are channel-dependent. We assume that the common control channel can support a transmission range larger than those associated with other channels. We model the network using a *communication graph* $G(V, E)$, where each node $v \in V$ corresponds to a mesh node and there is a link between nodes u and v ($u, v \in V$) if there exists at least a channel h available to both nodes and $||u - v|| \leq R_h$, where $||u - v||$ represents the Euclidean distance between u and v. Note that the topology of G depends on channel availability. So G needs to be re-constructed once there is a change. Two links e_1 and e_2 are said to interfere with each other if 1) e_1 is incident to e_2 (due to constraints enforced by half-duplexing, unicast communications or collisions) or 2) they work on the same channel h, and $||T(e_1) - T(e_2)|| \leq I_h$ or $||T(e_1) - R(e_2)|| \leq I_h$ or $||R(e_1) - T(e_2)|| \leq I_h$, $||R(e_1) - R(e_2)|| \leq I_h$, where $T(\cdot)$ and $R(\cdot)$ give the transmitter and the receiver of the given link respectively. Note that we use a symmetric interference model here because we assume that an ACK packet will be sent back to the transmitter by the receiver whenever it receives a data packet.

Due to channel availability changes, suppose that K end-to-end flows and a set E^f of links are affected. The recovery problem seeks a solution to re-allocate the available transmission blocks to the failed traffic such that the total number of fixed flows is maximized or the total amount of fixed end-to-end traffic is maximized. We will call them the *maximum Number of flows Recovery Problem (NRP)* and *maximum Throughput Recovery Problem (TRP)* respectively in the following. As mentioned before, we consider both single-hop and two-hop recovery schemes and assume all the failed traffic of flow k on link e will be re-routed if there is no feasible single-hop based solution. No resource re-allocation will be made to the unaffected traffic. The path of a flow may include multiple links, each of which may contain a different amount of failed traffic. We do not allow partial fixing for a flow, i.e., the failed traffic associated with a flow is either completely fixed or not fixed at all, since only fixing failed traffic on some of the links on the path of a flow may not improve its end-to-end throughput.

Before presenting the ILP formulation for the recovery problems, we summarize necessary notations as follows.

$f_{e,k}$: the amount of failed traffic of flow k on link e.

P_k: the set of links with failed traffic belonging to flow k.

$f_k = \max_{e \in P_k}\{f_{e,k}\}$.

K: the number of flows with failed traffic.

B_e: the set of IDs of free transmission blocks on link e, which can be identified in advance such that fixing failed traffic will not affect any existing non-failed traffic.

V_e: the set of common *usable* neighbors of nodes u and v, where $e = (u, v)$. A node $w \in V_e$ is *usable* if $B_{e_1} \neq \emptyset$ and $B_{e_2} \neq \emptyset$, where $e_1 = (u, w)$ and $e_2 = (w, v)$.

L_e^m: $0 \leq m \leq |V_e|$, L_e^0 only includes link e, and L_e^m ($m > 0$) include $e_1 = (u, w)$ and $e_2 = (w, v)$, where $e = (u, v)$ and w is the m^{th} node in V_e.

IE_e^h: the set of links interfering with link e on channel h.

$CB_{e,k}$: the set of *conditional free block* associated with flow k and link e. If a flow k cannot be fixed, then some transmission blocks originally allocated to a link e on the path of flow k can be released if $f_{e,k} < f_k$ since the traffic carried by these blocks cannot reach destination due to the bottleneck link (i.e., the link whose $f_{e,k} = f_k$). Such transmission blocks are called *conditional free blocks* since they are considered as free if the corresponding flow cannot be fixed.

$\rho_{e,k}^m$: a decision variable in the ILPs, where $0 \leq m \leq |V_e|$. $\rho_{e,k}^m = 1$ if some free blocks on link e are re-allocated for failed traffic of flow k on link e ($m = 0$) or the m^{th} ($m > 1$) node in V_e is used for re-routing; 0, otherwise.

$x_{e,k}^i$: a decision variable in the ILPs, where $1 \leq i \leq |B_e|$. $x_{e,k}^i = 1$ if free block i is re-allocated for failed traffic of flow k on link e; 0, otherwise.

y_k: a decision variable in the ILPs. $y_k = 1$ if all the failed traffic of flow k is fixed; 0, otherwise.

The ILP for the NRP is given in the following. The objective is to maximize the number of fixed flows. Constraint (2) guarantees that the total capacity of the transmission blocks allocated for fixing failed traffic of flow k on link e must be no small than the amount of failed traffic. According to Constraint (3), either the free blocks on link e, or one of the common usable neighbors of the two end nodes and the corresponding free blocks are used to fix the corresponding failed traffic. Constraint (4) establishes a connection between two variables, $\rho_{e,k}^m$ and y_k, and ensures that a failed flow is fixed if the traffic on all related links on its path is fixed. Constraint (5) indicates that no more than one free blocks with a common timeslot are allowed to be allocated to any link. Constraint (6) makes sure that a block will not be allocated to two interfering links. Constraint (7) ensures that a conditional free block corresponding to link e and flow k can only be used to fix other flows if flow k is not fixed. Similarly, we can formulate an ILP for the TRP whose objective is to maximize total amount of fixed end-to-end traffic. Note that by simply setting $|V_e|$ to $\emptyset$, we can restrict the solution to a single-hop based solution.

ILP - NRP

$$\max \sum_{1 \le k \le K} y_k \tag{1}$$

subject to:

$$\sum_{i \in B_{e'}} c_i x^i_{e',k} \ge f_{e,k} \rho^m_{e,k} \qquad \begin{array}{c} \forall e \in P_k, 1 \le k \le K, \\ 0 \le m \le |V_e|, \\ \forall e' \in L^m_e; \end{array} \tag{2}$$

$$\sum_{0 \le m \le |V_e|} \rho^m_{e,k} \le 1, \qquad \forall e \in P_k, 1 \le k \le K; \tag{3}$$

$$\sum_{e \in P_k} \sum_{0 \le m \le |V_e|} \rho^m_{e,k} = y_k \cdot |P_k|, \qquad 1 \le k \le K; \tag{4}$$

$$\sum_{i \in B_e | t_i = t} \sum_{1 \le k \le K} x^i_{e,k} \le 1, \qquad \forall e \in E^f, 1 \le t \le T; \tag{5}$$

$$x^i_{e_1,k_1} + x^i_{e_2,k_2} \le 1, \qquad \begin{array}{c} \forall e_1 \in E^f | i \in B_{e_1}, \\ \forall e_2 \in E^f | i \in B_{e_2}, \\ 1 \le k_1, k_2 \le K, \\ e_2 \in IE^{h_i}_{e_1}; \end{array} \tag{6}$$

$$x^i_{e',k'} \le 1 - y_k, \qquad \begin{array}{c} 1 \le k \le K, \forall e \in P_k \\ 1 \le k' \le K | k' \ne k \\ \forall i \in CB_{e,k}, e' \in IE^{h_i}_e. \end{array} \tag{7}$$

ILP - TRP

$$\max \sum_{1 \le k \le K} f_k y_k \tag{8}$$

subject to: Constraints (2) to (7)

IV. Distributed Recovery Algorithm

In this section, we describe a distributed algorithm for the recovery problems. For the purpose of transmission block re-allocation, every node needs to keep track of some necessary information of all the affected links in its interference neighborhood, such as failed traffic, free transmission blocks and so on. The end node of a link with smaller ID is considered as the head of that link. In our algorithm, each head node will broadcast these information to its two-hop neighbors over the common control channel using the control radio whenever there are any changes on channel availability. Since the transmission range that can be supported by the common control channel is larger than those associated with other channels, every head node usually can obtain the information of all the links which potentially interfere with its own links. In addition, whenever the channel availability changes, the head node of every link with failed traffic will also report related information to all the other nodes on the paths of the corresponding flows.

For simplicity, the failed traffic of flow k on link e is also called the failed traffic of *link-flow pair* (e, k). The algorithm is supposed to re-allocate free transmission blocks for the failed traffic of each affected link-flow pair. In order to solve the recovery problems in a distributed way, we have to order all the link-flow pairs carrying failed traffic in each interference neighborhood such that a free block will not be assigned to two interfering links. After a large number of experiments, we find out that ordering them in terms of the amount of failed traffic of a flow f_k and the amount of failed traffic of a link-flow pair $f_{e,k}$ offers the best performance. A link-flow pair (e', k') is said to be a predecessor of link-flow pair (e, k) if and only if *a*) $f_{k'} > f_k$; or *b*) $f_{k'} = f_k$, $k' < k$; or *c*) $f_{k'} = f_k$, $k' = k$, and $f_{e',k'} < f_{e,k}$; or *d*) $f_{k'} = f_k$, $k' = k$, $f_{e',k'} = f_{e,k}$, and $ID_{e'} < ID_e$. In *b*) and *c*) we break the ties using the IDs of the flows and the links respectively.

Every node v listens recovery notification messages from other nodes over the common control channel. Once it receives a message indicating the failed traffic of a link-flow pair (e', k') has been considered, it will check all the link-flow pairs associated with it. If it finds out that all the link-flow pairs (in its interference neighborhood) preceding one of its own link-flow pair (e, k) have been considered, it will use a subroutine to re-allocate free transmission blocks for failed traffic of (e, k) and broadcast a new recovery notification message in its interference neighborhood and to the other nodes on the path of flow k to indicate either the corresponding failed traffic is fixed (if there is a feasible solution) or this flow cannot be fixed (if no feasible solution).

In the following, we describe the subroutine which re-allocates free transmission blocks for link-flow pair (e, k). We assign a weight (between 0 and 1) to each candidate free block i on link e_c (e_c could be e or the links used for re-routing). The weight function $w(\cdot)$ is defined in equation (9), in which $f_{e,k}$ gives the amount of failed traffic of link-flow pair (e, k) and $\Gamma_{e'} = \sum_{i \in B_{e'}} c_i$ gives the total capacity of all free blocks on link e'.

$$w(i, e_c) = \alpha \max \left\{ 0, 1 - \frac{c_i}{f_{e,k}} \right\} + (1 - \alpha) \max_{e' \in IE^{h_i}_{e_c} | i \in B_{e'}} \left\{ \frac{c_i}{\Gamma_{e'}} \right\} \tag{9}$$

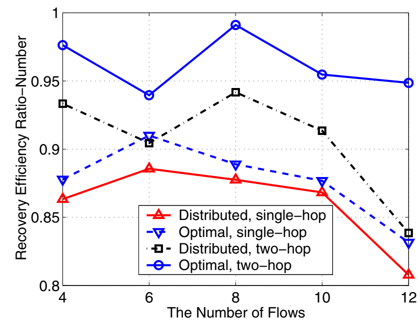
In the first term, $\frac{c_i}{f_{e,k}}$ indicates how much failed traffic of link-flow pair (e, k) can be fixed if the candidate block i is chosen. The second term indicates the impact of selecting block i on its interference neighborhood. A relatively large value indicates that the total capacity of free blocks (other than block i) on the links which also have block i as free blocks in the interference neighborhood is fairly small, i.e., if block i is chosen, those links may not have sufficient resources to fix other failed traffic, which is obviously not preferred. On the contrary, a relatively small value indicates that selecting block i would not seriously affect the availability of free blocks on the neighboring links. Overall, a transmission block with a relatively small weight value is preferred to be selected. After conducting extensive simulations, we find out that $\alpha = 0.5$

offers the best performance.

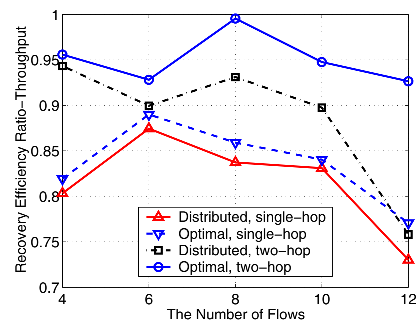
According to the algorithm, each node will first consider a single-hop based solution for link-flow pair (e, k), i.e., select free transmission blocks on link e to fix the failed traffic. The free blocks will be sorted in the ascending order of its weights and will be considered as part of the solution one by one such that the failed traffic $f_{e,k}$ is completely fixed and two blocks with a common timeslot will not co-exist in the solution. If no feasible solution can be obtained, the algorithm will try to select a common neighbor to re-route the failed traffic. The common neighbors are considered in the ascending order of their IDs. Similarly, free blocks will be sorted in the ascending order of their weights and will be considered as part of the solution one by one such that the failed traffic $f_{e,k}$ is completely fixed (the total capacity of allocated blocks on each link used for re-routing should be no less than $f_{e,k}$) and the interference constraints are not violated. After this allocation procedure, a message will be broadcast to nodes in the interference neighborhood and the other nodes on the path of flow k to notify the recovery solution or the failure of recovery attempt. Note that if the algorithm decides that there is no feasible solution for a failed flow, the corresponding conditional free blocks will be released to fix other flows.

V. Simulation Results

In this section, we evaluate the performance of the proposed schemes via simulations using NS2 [7] and CPLEX 10.1 [3]. In the simulation, mesh nodes were randomly placed in a 1000m $\times$ 1000m square region. All the channels were divided into three groups, each of which has the same number of channels. The channels in the first group have a transmission range of 250m and a link data rate of 11Mbps. The values were set to 200m and 22Mbps for the channels in the second group, and 150m and 54Mbps for the channels in the third group. On every channel, the interference range was set to 2 times the corresponding transmission range. Every primary user was randomly placed and its channel was randomly selected. We set the number of channels to 12, the number of timeslots in a frame to 20, and the number of active primary users to 8. In the simulation, we generated the flows and traffic as follows: for each flow, we randomly chose a source, a destination, and a traffic demand in the range $[0.1, 5]$Mbps. We then applied the Dijkstra's algorithm to find a shortest path and a simple random algorithm to allocate free transmission blocks for each link along the path to satisfy its traffic demand subject to the interference constraint. The flow was accepted only if a feasible routing and block allocation solution can be found. Otherwise, a new flow will be generated and its feasibility will be checked. After this process, we created a set of end-to-end flows and identified the used and free transmission blocks on each link. The ratio between the number of fixed flows and the total number of failed flows, and the ratio between the amount of fixed traffic and the total amount of failed traffic are used as performance metrics and they are referred to as "recovery efficiency ratio-number" and "recovery efficiency ratio-throughput" respectively. Note that the optimal "number" and "throughput" values were obtained by solving ILP-NRP and ILP-TRP respectively. Each number presented in the figures are the average over 30 runs. In each run, a network and a set of flows were randomly generated as described before.



(a) Recovery efficiency ratio-number



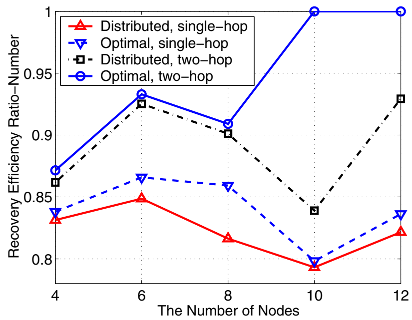
(b) Recovery efficiency ratio-throughput

Fig. 1. The distributed algorithm VS. the optimal solutions in terms of different numbers of flows

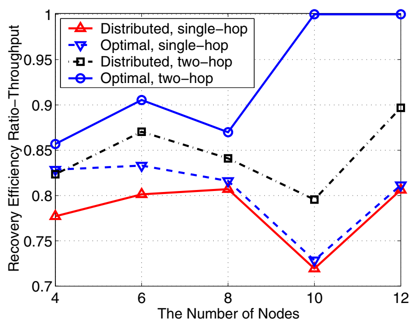
In the first two scenarios, we compared the distributed algorithm with the optimal solutions given by solving the ILPs. In scenario 1, we changed the number of end-to-end flows from 4 to 12 with the number of mesh nodes (network size) fixed to 10. The corresponding results are presented in Fig. 1. In scenario 2, we changed the network size from 4 to 12 with the number of flows fixed to 8. The corresponding results are presented in Fig. 2. In scenario 3, we evaluated the performance of the distributed algorithm in large networks with 40 mesh nodes. We changed the number of flows from 10 to 70. The results are shown in Fig. 3. We make the following observations from the results:

1) Figs. 1 and 2 show that the performance given by our distributed algorithm is very close to that of the optimal solutions. On average, the distributed algorithm and solving the ILP-NRP achieve a recovery efficiency ratio-number of 87.0% and 90.5% respectively. Similarly, the distributed algorithm and solving ILP-TRP give an average recovery efficiency ratio-throughput of 83.2% and 87.9% respectively. The differences are only 3.5% (number) and 4.7% (throughput) respectively.

2) From the figures, we find out that the recovery ratios do not monotonically increase with the number of flows. Intuitively, increasing the number of flows will introduce more failed traffic, which could lead to lower recovery ratios. However, the resources available for recovery actually include

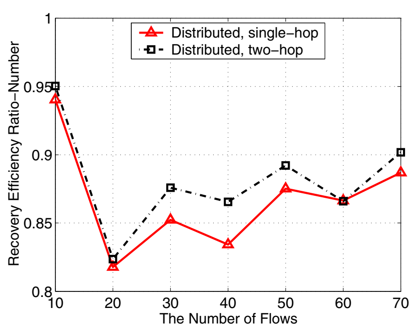


(a) Recovery efficiency ratio-number

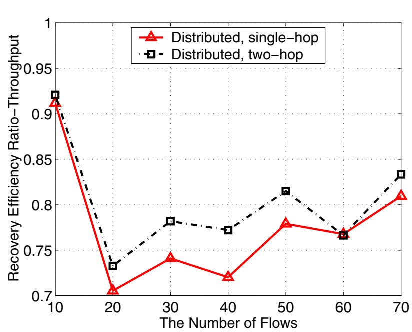


(b) Recovery efficiency ratio-throughput

Fig. 2. The distributed algorithm VS. the optimal solutions in terms of different network sizes



(a) Recovery efficiency ratio-number



(b) Recovery efficiency ratio-throughput

Fig. 3. The performance the distributed algorithm in large networks

two types of transmission blocks: free and conditional free. More failed flows may result in more conditional free blocks, which can help fix more failed traffic. In addition, we can also see that the recovery ratios do not monotonically increase with the network size either. On one hand, increasing the networks size will introduce more failed flows as well as stronger interference, which are not favorable for recovery. On the other hand, more mesh nodes will bring more available resources and more options for traffic re-routing, which can help improve the recovery efficiency ratios.

3) As expected, the two-hop scheme outperforms the single-hop scheme by 6.0% in terms of recovery efficiency ratio-number and by 7.6% in terms of recovery efficiency ratio-throughput on average. This is easy to understand since the two-hop scheme allows traffic re-routing.

VI. Conclusions

In this paper, we studied recovery from failures caused by channel availability changes in WMNs with cognitive radios. We formulated the corresponding problems as ILP problems and solved them to obtain optimal solutions. Moreover, we proposed an efficient distributed algorithm for fast recovery. Simulation results showed that the performance given by our distributed algorithm is very close to that of the optimal solutions.

References

[1] I. F. Akyildiz, X. Wang and W. Wang, Wireless mesh networks: a survey, *Journal of Computer Networks*, Vol. 47, No. 4, 2005, pp. 445–487.

[2] I. F. Akyildiz, W-Y Lee, M. C. Vuran and S. Mohanty, NeXt generation/dynamic spectrum access/cognitive radio wireless networks: a survey, *Journal of Computer Networks*, Vol. 50, No. 13, 2007, pp. 2127–2159.

[3] ILOG Software Inc., CPLEX 9.0, http://www.ilog.com

[4] P. Gupta and P. R. Kumar, The capacity of wireless networks, *IEEE Transactions on Information Theory*, Vol. 46, No. 2, 2000, pp. 388–404.

[5] R. Hincapie, J. Tang, G. Xue and R. Bustamante, QoS routing in wireless mesh networks with cognitive radios, *Proceedings of IEEE Globecom'2008*.

[6] Y. T. Hou, Y. Shi and H. D. Sherali, Optimal spectrum sharing for multi-hop software defined radio networks, *Proceedings of IEEE Infocom'2007*, pp. 1–9.

[7] Network Simulator 2 (NS2), http://www.isi.edu/nsnam/ns/

[8] A. Raniwala and T. Chiueh, Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network, *Proceedings of IEEE Infocom'2005*, pp. 2223–2234.

[9] J. Tang, G. Xue and W. Zhang, Interference-aware topology control and QoS routing in multi-channel wireless mesh networks, *Proceedings of ACM MobiHoc'2005*, pp. 68–77.

[10] J. Wang, Y. Fang and D. Wu, A power-saving multi-radio multi-channel MAC protocol for wireless local area networks, *Proceedings of IEEE Infocom'2006*.

[11] Y. Xi and E. M. Yeh, Distributed algorithms for spectrum allocation, power control, routing, and congestion control in wireless networks, *Proceedings of ACM MobiHoc'2007*, pp. 180–189.

[12] C. Xin, B. Xie, C-C. Shen, *A novel layered graph model for topology formation and routing in dynamic spectrum access networks*, *Proceedings of IEEE DySPAN'2005*, pp. 308-317.

[13] Y. Yuan, P. Bahl, R. Chandra, T. Moscibroda and Y. Wu, Allocating dynamic time-spectrum blocks in cognitive radio networks, *Proceedings of ACM MobiHoc'2007*, pp. 130-139.

[14] H. Zheng and C. Peng, Collaboration and fairness in opportunistic spectrum access, *Proceedings of IEEE ICC'2005*, pp. 3132-3136.