Channel Dynamics Matter: Forwarding Node Set Selection in Cognitive Radio Networks

Ying Dai, Jie Wu, and Andrew Daniels
Department of Computer and Information Sciences, Temple University, Philadelphia, PA 19122
Email: {ying.dai, jiewu, andrew.s.daniels}@temple.edu

Abstract—**Opportunistic routing is very promising when applied to cognitive radio networks (CRNs). However, this requires solving the forwarding node (FN) set selection problem for each node in CRNs. The FN set selection is different from that in traditional networks because the reliability under suddenly active primary users (PUs) and the co-channel interference have to be taken into account. Moreover, the selected FN set cannot stay static in CRNs due to the dynamic channel environment. It means that an adjustment scheme of the FN set is necessary. We construct a FN set selection model for CRNs, which prioritizes each FN candidate considering not only its ETX metrics, but also the channel dynamics and co-channel interference. Based on the defined priorities of each node, we propose three algorithms, a basic greedy algorithm, a greedy algorithm with one backtrack, and a maximum weighted independent set (MWIS) algorithm, for FN set selection aiming at different performance requirements. Extensive simulations are performed to evaluate our FN set selection algorithms.**

Index Terms—**Cognitive radio networks, forwarding node set selection, opportunistic routing.**

I. INTRODUCTION

Cognitive radio networks (CRNs) [1] enable each secondary user (SU), or node, to make opportunistic use of channels when they are not occupied by primary users (PUs). When PUs become active, the nodes occupying channels of the PUs within the same area must leave those channels. Since the activities of PUs are unpredictable under most circumstances, it leads to the channel dynamics in CRNs, which pose challenges to many issues in CRNs. One of them is the routing issue.

A very promising solution to the routing problem in CRNs is the opportunistic routing [2]–[4], since nodes in such opportunistic routing protocols do not stick to a particular route. Instead, a sender broadcasts its data. Nodes which hear the transmission and are closest to the destination will forward the data. This scheme is particularly useful for routing in CRNs, considering the special interference and channel dynamics. In CRNs, different from other wireless networks, the interference to a single link can come from PUs and SUs. If the interference is caused by SUs, the interfered nodes can compete for channel access. However, if the interference is caused by PUs, the nodes that have been interfered with cannot compete with PUs, but quit from that channel. This causes links in CRNs more vulnerable and taking a very long time to recover once interfered by PUs. Therefore, routes consisting of single links in CRNs are unreliable when facing the unpredictable PU activities. Since the opportunistic routing does not necessarily rely on a single route, which is more reliable under the dynamic channel availabilities, it can potentially be very useful if applied in CRNs. With tremendous works on opportunistic routing problems in other wireless networks, it brings up a question: are the previous approaches of opportunistic routing directly applicable on CRNs?

In fact, the answer is negative. One of the problems is the forwarding node (FN) set selection of opportunistic routing in CRNs. The FN set selection in traditional wireless networks usually relies on the distance to destinations, link qualities, estimation of delivery probabilities, and so on. There are no PUs in traditional wireless networks, which means nodes in these networks do not need to stop using channels immediately for active PUs. Therefore, the FN set selection scheme in traditional wireless networks has no consideration for PUs. However, in CRNs, the FN set selection has to consider the channel dynamics caused by PUs. Nodes with more available channels are more reliable during transmission when facing the suddenly active PUs. It means that the FN set selection cannot be limited to the link quality, but also needs to consider the channel dynamics in CRNs. Moreover, the co-channel interference between links in the same interference area needs to be considered. As a result, when selecting FN sets in CRNs, the criteria should take the dynamic channel availabilities and possible channel interference into consideration.

In this paper, we propose a novel FN set selection model in CRNs, which aims at meeting the requirements regarding the delivery rate. We consider both the reliability and the co-channel interference, and define the weight of each candidate FN. For different performance considerations, we propose three algorithms to select the FN set for each node: a basic greedy algorithm, a greedy algorithm with one backtrack, and a maximum weighted independent set (MWIS) algorithm.

The main contributions of our work are as follows:

- To our best knowledge, this is the first work to consider the FN set selection in CRNs, under the opportunistic routing.
- We define the weight of each candidate FN by taking the dynamic channel availabilities and co-channel interference into account.
- We propose three algorithms for the FN set selection, aiming at different performance requirements with varied complexities.

The organization of our paper is as follows. In Section II, we discuss the related works. The problem formulation is introduced in Section III, which describes the model and defines our problem. The FN set selection model is presented in Section IV, which contains the FN prioritization, three FN selection algorithms. The performance evaluations of our FN selection algorithms are given in Section V. We conclude our paper in Section VI.

II. RELATED WORKS

Since there have been many works on opportunistic routing in wireless networks, in this section, we discuss the existing forwarding node set selection algorithms and the problems of directly applying them in CRNs.

1) Classical Forwarding Node Selection Algorithms: Works in [2]–[4] are among the well-known opportunistic routing protocols. In the ExOR protocol [2], the forwarding node list is chosen based on the estimated transmission count (ETX) [5] of each node to the destination. Also, the source only chooses a subset of nodes in the forwarder list for forwarding, due to the consideration that if too many nodes are forwarding the data, the expected number of a batch's packet on each node may be close to zero. In [3], the authors propose MORE, which efficiently apply network coding in opportunistic routing. The forwarding node set selection is similar to the method in ExOR. The difference is that the forwarders in MORE only transmit the innovative packets, which are linearly independent packets from what the forwarders have received, since every packet is performed with network coding. The forwarding node selections in both ExOR and MORE do not consider or prevent diverging paths. SOAR in [4] selects the FNs and prevents diverging paths by putting ETX constraints on the FNs and nodes of the default paths.

2) Forwarding Node Selection in Different Wireless Networks: Many works have been done related to the FN set selection in other wireless networks [6]–[9]. A pressure routing protocol for underwater sensor networks is proposed in [6]. It considers simultaneous packet receptions among one's neighbors, which enables the opportunistic forwarding by a subset of the neighbors that have received the packet correctly. The authors consider the FN set selection, which takes the channel quality into account. They also propose a simple greedy heuristic approach that searches for a cluster of nodes with maximum progress and without hidden terminal problems, using local topology and geographical information. Two relay node selection schemes are proposed in [8], [9] for mobile ad hoc networks. CORMAN in [7] is an extension of ExOR in mobile ad hoc networks. Every time a packet is received by a downstream node, it starts from that node and reaches the destination node earlier. The approach in [8] is based on the estimation of the future path. The model in [9] is based on the sharing of geographical locations using GPS. None of the above approaches can be applied on the FN set selection in CRNs, due to channel dynamics and unstable links.

3) Specialities of Forwarding Node Selection in CRNs: The FN set selection in CRNs cannot directly adopt the approaches above. The selection criteria based on ETX, link quality, and channel quality is not sufficient in CRNs. This is because the PU activities in CRNs are unknown and unpredictable. Without the prediction of PU activities and PU interference on each channel, the performance metrics of ETX, link quality, channel quality, and so on are not precise, due to the reason that the sudden appearance of PUs could cause links to be broken and unrecoverable if the PU activities last very long. Therefore, our work makes use of the previous works, and considers the specialities of CRNs to select the FN set.

III. PROBLEM FORMULATION

We consider that, in a CRN with node set $\{u, v, w, ...\}$, the total available channel set is M. There are also a number of PUs with random probabilities of being active. When the PUs become active on certain channels, the nodes within the PUs' interference area have to quit the channels. For a node u, we use M_u to denote the current available channel set of u. The sender and receiver have to tune to the same channel to communicate. We assume there is a common control channel (CCC) in our model. The CCC assumption is for the simplicity concern, but is not necessary if we apply the control channel calculation approach in [10]. In addition, the SINR(signal to interference-plus-noise ratio) from the sender node to the receiver has to be large enough for a successful transmission.

Assume that the opportunistic routing is applied for the data transmission. We are not limited to a specific opportunistic routing protocol. Our model is to add the consideration of specialities in CRNs to the FN selection. Suppose we apply SOAR [4] here, and each node maintains a routing table containing (destination, default path, and FN set). Our goal is to choose the FN set from the neighbor set of each node, and ensure that the delivery rate to the destination node satisfies the predefined threshold. In the following parts, we formulate the delivery probability for a single link based on the above constraints, and provides the assurance for the dynamic channel adaptability.

For a single link between a pair of nodes with distance d transmitting on channel m ($m \in M$), the average SINR value is denoted as $\Gamma(d, m)$. We use the Rayleigh fading model to estimate small scale fading here [6]. When channel m is available, which means the PUs on m are inactive, the probability of bit error over a distance d is:

$$p_e(d, m) = \frac{1}{2}(1 - \sqrt{\frac{\Gamma(d, m)}{1 + \Gamma(d, m)}}). \tag{1}$$

In our model, we assume each packet has a constant length of L. The probability of PUs, whose interference areas contain this link, being inactive on channel m is $H(m)$. We use uv to denote the single link from node u to v, and d_{uv} to denote the distance between u and v. The channel set that can be used for transmission is $M_{uv} = M_u \bigcap M_v$. The probability of a packet being transmitted successfully over link uv can be written as:

$$p(d_{uv}) = 1 - \prod_{m \in M_{uv}} (1 - H(m) \cdot (1 - p_e(d, m))^L). \tag{2}$$

The $p(d_{uv})$ is related to both the SINR constraint, and the PU protection requirements in CRNs. It is only an estimation because $H(m)$ is a historically statistical value, and sometimes it is not precise. In addition, the interference of other links to uv is not static, since they may switch among different channels during the transmission. However, it does provide a good insight for the quality of a single channel, which will be one of the factors that are considered for FN set selection.

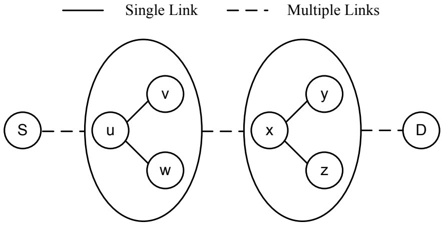


Fig. 1. The FN adjustment is needed for x, rather than u.

From the above discussions, it is impractical to find an optimal FN set for each node, due to the dynamic channel environment. We propose a practical and effective FN set selection model, which considers both the co-channel interference among links within the same interference area, and the reliability when PUs become active during transmission. Also, an efficient FN adjustment scheme is necessary to ensure the delivery performance of opportunistic routing in CRNs.

IV. FN SET SELECTION MODEL

A. Model Overview

For a node u, the overview of its FN set selection is :

- Node u selects a subset from its neighbor set, using our FN set selection algorithm;
- Having the FN set selected, node u sends the coded packets on the channel coordinated using CCC;
- Once PUs appear and cause the channel being used by node u to become inactive, u would switch to another available channel and continue transmitting.

Here are some illustrations. The FN set selection algorithm is described in the following parts. We propose three algorithms, and each of them is suitable for different scenarios. Nodes in the FN set of u will receive packets from u using the original channel as u. The FN set selection algorithms do not depend on which channel is used by u for transmission. The channel selection and coordination among multiple nodes for packet broadcasting, which can be conducted by transmission performance estimation and controlling message exchanges, is beyond the scope of this paper. When more PUs become active, and the channel switching cannot meet the delivery rate requirement, the FN set of some nodes will be adjusted.

In the following parts, we first prioritize the candidate FN nodes in the neighbor set by defining their weight, considering the channel dynamics in CRNs. Then, three FN set selection algorithms are proposed and compared.

B. Forwarding Node Prioritization

The delivery probability estimation of a packet over a single link is given in Eq. 2. From Eq. 2, it is obviously that the main factors on the delivery probability are:

1) the interference, $I(m)$, from other links within the interference area of uv that are also working on channel m;
2) the probability of PUs, $H(m)$, whose interference area contains link uv, that are inactive on channel m.

Therefore, both factors should be taken into consideration when selecting FNs from neighbor nodes.

For simplicity, we assume that the interference range of a single link uv is two-hop, which includes all adjacent links of uv. This means that $I(m) = \infty$ when there is an adjacent link also using m; $I(m) = 0$, otherwise. This assumption is not a requisite for our model. Other interference estimation methods can be easily applied here.

Assume that u needs to select FN from its neighbor set, N_u. For a node $v \in N_u$. M_u is the available channel set of node u. We use M_{uv} to denote the channels available on link uv, and $M_{uv} = M_u \bigcap M_v$. Considering the interference from adjacent links, for any channel $m \in M_{uv}$, the conflict probability of adjacent links choosing the same m as uv is defined as [11]:

$$C_{uv}(m) = \sum_{w \in N_v} \frac{1}{|M_{uw}|} E_{uw}(m) + \sum_{w \in N_u} \frac{1}{|M_{vw}|} E_{vw}(m), \tag{3}$$

where $E_{uw}(m)$ is a step function with a value of 1 when $m \in m_{uw}$ and is 0 otherwise. Then, we define "receiving ability" regarding a neighbor node v of node u.

***Definition** 1:* For $\forall v \in N_u$, the receiving ability of v from u is:

$$R_{uv} = \sum_{m \in M_{uv}} \frac{1}{C_{uv}(m)}. \tag{4}$$

This definition considers the interference from the adjacent links. Node v, with more channels that are less likely to be used by adjacent links, has a better value of R_{uv}. Moreover, node v that has more available channels to be used for receiving from u is more reliable. When PUs nearby suddenly become active, it can switch to one of the other channels, and continue transmission.

However, it is not enough to prioritize each neighbor node using the notion of receiving ability. This is because node v with a better value of R_{uv} is likely to have a worse ability of forwarding to the next-hop nodes of v. When node u decides to choose v as its FN, it also needs to estimate the forwarding ability of v. Assume that u has the two-hop information when it performs the estimation. We have the following definition of the weight of node v to be chosen as a FN of u:

***Definition** 2:* For $\forall v \in N_u$, the weight of v to be selected as a FN of u is:

$$W_{uv} = R_{uv}\Big(\sum_{w \in N_v} |M_{vw} - M_{uv}|\Big). \tag{5}$$

$|M_{vw} - M_{uv}|$ is the number of elements left in M_{vw} after removing the elements in M_{uv} from M_{vw}. With the same receiving ability, the nodes that have more non-conflict channels with u and more links to forward the packets will have a larger weight.

Here, if v can be selected as the FN of u, it should be closer in terms of its proximity to the destination. The proximity here is not measured by physical distance, but the ETX metrics as in [5]. Additional constraints of ETX can be applied here, e.g., the four constraints in SOAR [4]. Our FN node set selection algorithm can be combined with other FN selection algorithms in terms of different proximity metrics.

Algorithm 1 Basic Greedy Algorithm to Calculate F_u of u

```
$N'_u$ is the list to store the FN candidates
for $v \in N_u$ & $v$ is smaller of ETX to the destination than $u$ do
    if ETX of $v$ to the destination $< \alpha$ then
        Calculate $W_{uv}$ using Eq. 5
        Insert $v$ to $N'_u$
while $N'_u$ is not empty do
    Set $v$ as the node with the max $W_{uv}$ in $N'_u$
    if $d_{vw} > \sigma$, $\forall w \in F_u$ then
        $F_u = F_u + \{v\}$
    $N'_u = N'_u - \{v\}$
```

C. FN Set Selection Algorithms

Having the weight of each neighbor defined, we propose three algorithms of the FN set selection for each node.

1) Basic Greedy Algorithm: Intuitively, the easiest way to select a FN set is to apply the greedy algorithm. For a node u, the basic greedy algorithm of choosing its FN set, F_u, is described in Algorithm 1. The FN set is selected from the downstream neighbors of u, which are closer to the destination than u, in terms of ETX metrics, and also satisfy the ETX threshold α. The ETX here of a single link is calculated as $\frac{1}{(1-p_r)\times 1-p_f}$, where p_r and p_f denote the loss probabilities of the link in the forward and reverse directions, respectively. The ETX metrics have been proved useful in [12]. Also, we set the distance constraint σ on nodes in F_u. There are two reasons for setting σ on the FN set. First, nodes in close geographical locations share similar channel availabilities in CRNs. If two nodes are very close together, it is possible that they choose the same channel for data transmission, which would cause co-channel interference. Second, when a PU becomes active, nodes that are too close to each other may be affected together, and they all need to switch channels. Therefore, we require nodes in the the FN set to satisfy the minimum distance requirements. Having the ETX and distance constraints satisfied, the node with the maximum weight is selected, and it will be added to F_u.

This greedy algorithm can provide a relatively reliable FN set with low complexities. Since the weight of each node is defined based on the overall channel availabilities, instead of one specific channel, u would broadcast to its F_u using one of its available channels. When PUs suddenly appear and affect the delivery rate, u can gain this knowledge based on the ACKs, and switch to another channel to broadcast to F_u. When the channel switching cannot satisfy the requirements of delivery rate, u needs to rerun this algorithm and find a new F_u. Our algorithm reduces the number of times to reselect the FN set, since the selected nodes are more adaptable to dynamic channel availabilities.

2) Greedy Algorithm With One Backtrack: The basic greedy algorithm is very straightforward and easy to implement. However, it is likely that the FN set selected by the greedy algorithm is too small. For example, under some circumstances, once the node with the maximum weight is selected, there is only one more, or even no nodes that satisfy both the ETX and distance requirements, which makes the size of the FN set only contain one or two elements. This would harm the routing performance since the forwarding ability of FN set is harmed. Also, if the application scenario has other requirements on the FN set, the greedy algorithm is very likely to fail in satisfying the requirements. Therefore, the greedy algorithm is likely to be ineffective. To deal with this situation, we provide one backtrack scheme for the greedy algorithm.

Algorithm 2 Greedy Algorithm With Backtrack List Maintained to Calculate F_u of u

```
$L_B$ is the list to store the backtrack nodes
$N'_u$ is the list to store the FN candidates
for $v \in N_u$ & $v$ is smaller of ETX to the destination than $u$ do
    if ETX of $v$ to the destination $< \alpha$ then
        Calculate $W_{uv}$ using Eq. 5
        Insert $v$ to $N'_u$
while $N'_u$ is not empty do
    Set $v$ as the node with the max $W_{uv}$ in $N'_u$
    if $d_{vw} > \sigma$, $\forall w \in F_u$ then
        $F_u = F_u + \{v\}$
    Set $v'$ the node with the second max $W_{uv'}$ in $N'_u$
    if $d_{v'w} > \sigma$, $\forall w \in F_u$ then
        $L_B = L_B + \{v'\}$
    $N'_u = N'_u - \{v\}$
```

During every loop in which one node is selected into the FN set by the greedy algorithm, we keep a backtrack node for it. For example, if node v is selected into the FN set, we would select a node v' with the second largest weight. If the final result of the FN set selected by the greedy algorithm cannot meet the other requirements (e.g., the size requirement of FN set), our backtrack scheme will go back one step, replace one element every time, and rerun the greedy algorithm from that point. The details of the greedy algorithm for selecting F_u for node u with backtrack list L_B maintained is in Algorithm 2. The initial FN set selection process is similar to the greedy algorithm, except that a L_B list is maintained to store the backtrack node for every node in the initial F_u. Here, some overlap is possible between the two lists, F_u and L_B.

The backtrack algorithm is in Algorithm 3 with the size requirement τ. Similarly, it can be extended for other requirements other than τ. When the F_u cannot meet the requirement τ, the FN set would be updated by removing all the nodes with weights less than the one pointed by ptr_1. Also, the ptr_1 and ptr_2 are used to maintain the backtrack node. Then the greedy algorithm will be applied to select new nodes to the FN set from the remaining nodes with weights less than that of the node pointed by ptr_2. The process will continue until the FN set meets the constraint of τ. We only maintain one backtrack list here. Of course, more backtrack lists can be maintained, if one cannot find the appropriate FN set. The complexity of this backtrack scheme is low, since only one node is maintained for each node in the FN set.

3) Maximum Weighted Independent Set Algorithm: The basic greedy algorithm and the greedy algorithm with one back track cannot ensure that the selected FN set is the one with the maximum weight. Here, under the same ETX and distance constraints, we propose another algorithm to give the optimal result as of the overall weight. For the neighbor set N_u of node u, we construct a graph $G_u(\sigma)$ from nodes that satisfy

Algorithm 3 Backtrack Algorithm With Size Constraint τ

```
Pointer $ptr_1$ points to the end of the $F_u$
Pointer $ptr_2$ points to the end of the $L_B$
$N'_u$ is the list to store the FN candidates
while $|F_u| < \tau$ do
    Remove all nodes with weights less than that of the node pointed by $ptr1$
    Replace the node of $F_u$ that $ptr_1$ points to with the node pointed by $ptr_2$ in $L_B$
    for $v \in N_u$ & $v$ is smaller of ETX to the destination than $u$ do
        if ETX of $v$ to the destination $< \alpha$ then
            Calculate $W_{uv}$ using Eq. 5
            if $W_{uv} <$ the weight of the node pointed by $ptr1$ then
                Insert $v$ to $N'_u$
    while $N'_u$ is not empty do
        Set $v$ as the node with the max $W_{uv}$ in $N'_u$
        if $d_{vw} > \sigma, \forall w \in F_u$ then
            $F_u = F_u + \{v\}$
        $N'_u = N'_u - \{v\}$
    Move $ptr_1$, $ptr_2$ one step forward
```

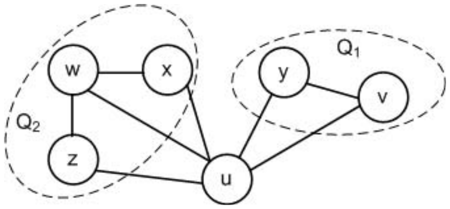


Fig. 2. An example of the modules for node u.

the ETX constraints α, defined as follows:

***Definition** 3:* Given node u, its neighbor set N_u, and the distance threshold σ, we define a graph $G_u(\sigma)$, where

1) v is a vertex in $G_u(\sigma)$, iff $v \in N_u$, v is smaller of ETX to the destination than u and satisfies ETX constraint α;
2) an edge exists between two vertices, v and w, in $G_u(\sigma)$ iff $d_{vw} > \sigma$,

Based on the weight definition in Eq. 5, the FN set selection does not rely on any specific channel used by each node. The goal here is to find the maximum weight set with the ETX and distance constraints. We define the independency here on the distance threshold σ, and convert the FN set selection to find the Maximum Weighted Independent Set (MWIS) of $G_u(\sigma)$.

The MWIS problem is a well-known NP hard problem. We adopt a recursive approach for MWIS calculation, based on module decomposition [13]. A module is defined as:

***Definition** 4:* Given a $G_u(\sigma)$, suppose U is a subset of the vertex in $G_u(\sigma)$. For a node v, which is a vertex of $G_u(\sigma)$ and $x \notin U$, x "distinguishes" U if x has both a neighbor and a non-neighbor in U. U is a *module* if it is indistinguishable for the vertices outside U.

We use $\{Q_k\}$ to denote the set of modules in $G_u(\sigma)$. An example is shown in Fig 2. Suppose $\{x, y, z, v, w\}$ are u's neighbors, and construct $G_u(\sigma)$. The node x distinguishes the node set $\{z, w\}$, since x has both a neighbor and a non-neighbor in $\{z, w\}$. x cannot distinguish the node set $\{y, v\}$ since neither y nor v is x's neighbor. There are two modules in this graph, which is $Q_1 = \{y, v\}$ and $Q_2 = \{z, w, x\}$.

Algorithm 4 MWIS Algorithm to Calculate F_u of u

```
1.  Set $G_u(\sigma)$ as empty
2.  if $G_u(\sigma)$ only has one vertex, $v$ then
3.      Return $F_u = \{v\}$ and stop
    // Construct the graph $G_u(\sigma)$
4.  for every $v \in N_u$ & $v$ is smaller of ETX to the destination than $u$ do
5.      Calculate $W_{uv}$ using Eq. 5
6.      Add $v$ to the vertex set of $G_u(\sigma)$
7.      for every $w$ that is a vertex in $G_u(\sigma)$ do
8.          if $d_{wv} > \sigma$ & ETX of $v$ to the destination $< \alpha$ then
9.              Add edge $wv$ to $G_u(\sigma)$
10. Divide $G_u(\sigma)$ into $\{Q_k\}$ using Definition 5.
11. for every $Q_k \in \{Q_k\}$ do
12.     Set $G_u(\sigma) = Q_k$
13.     Rerun from 2, with output denoted as $F_u^k$
14.     $F_u = F_u \bigcup F_u^k$
```

Having the module defined, the MWIS algorithm contains two steps: 1) decompose the $G_u(\sigma)$ into different modules; 2) recursively find the MWIS in each module to get the MWIS in $G_u(\sigma)$. The following are implementations of the two steps.

***Definition** 5:* Given a $G_u(\sigma)$, based on the fact whether it is connected or not, the module decomposition process is:

- if $G_u(\sigma)$ is disconnected, it can be divided into modules $\{Q_k\}$, which are connected components;
- if the complement graph of $G_u(\sigma)$, denoted as $\overline{G}_u(\sigma)$ is disconnected, $\overline{G}_u(\sigma)$ can be divided into $\{Q_k\}$, which are connected components;
- if both $G_u(\sigma)$ and $\overline{G}_u(\sigma)$ are connected, divide $G_u(\sigma)$ into maximal modules $\{Q_k\}$.

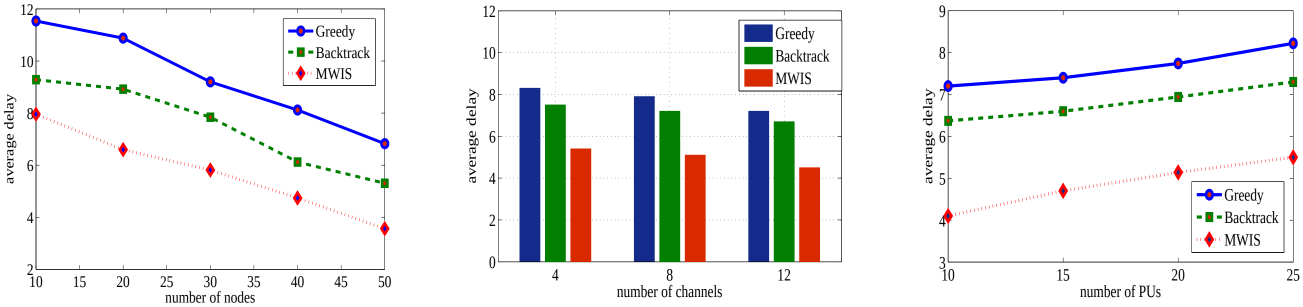
Obviously, for any module Q_k in connected $G_u(\sigma)$, there exists no node u outside Q_k that has both a neighbor and a non-neighbor in Q_k. The maximal modules in the third case are all pairwise disjoint.

Having the decomposition scheme, the recursive MWIS algorithm for finding FN set with the distance threshold σ is shown in Algorithm 4. The first part is to construct the $G_u(\sigma)$ based on the threshold σ. The second part is to divide the original $G_u(\sigma)$ into different modules or components $\{Q_k\}$. The third part is to recursively find the MWIS in each Q_k and return the MWIS of $G_u(\sigma)$.

V. PERFORMANCE EVALUATION

A. Simulation Settings

We randomly distribute nodes in a 200×200 unit square. There are three network parameters: the number of nodes, the number of total channels, and the number of PUs. The total number of node varies from 10 to 50. The communication range of each node is $[50, 70]$. Each node has its own set of



(a) change nodes

(b) change channels

(c) change primary users

Fig. 3. Comparison of delay under different network environment parameters.

available channels, depending on the position of PUs. There are $[4, 12]$ total channels, and the total number of PUs ranges from 10 to 25. Each PU has a probability 0.5 of being active at the beginning of each time slot, and occupies one channel. We assume each packet is transmitted at the beginning of each time slot. For a single link, the time cost to transmit a packet on an available channel is one time slot. We assign the loss rate of a single link as $[0.2, 0.8]$. By varying the three network parameters, we compare the average delay: given constant traffic requirements, the average value of delay used by each algorithm to reach the destination node.

B. *Simulation Results*

We compare the average delay by varying three different network parameters. In Fig. 3(a), the number of nodes varies from 10 to 50. The number of total channels is set to be 10, and the number of PUs is 15. The three lines denote the greedy algorithm, greedy algorithm with one backtrack, and MWIS algorithm. The delay is measured in seconds. The results show that MWIS has the least delay among the three algorithms, while the greedy algorithm has the most delay. The delay of all three algorithms decreases as the number of nodes increases. This is because more nodes can be selected in the FN set, given a certain traffic.

In Fig. 3(b), the average delay is compared by varying the number of total channels in the network. The total number of channels is varied from 4 to 12. The number of nodes is 25, and the number of PUs is 15. Again, the MWIS algorithm achieves the least delay among the three, and greedy algorithm with one backtrack is the second best. The delay of all three algorithms decrease as the number of channels increases. This is because more channels bring more channel opportunities for each link, and make each link more stable.

In Fig. 3(c), we vary the total number of PUs in the network, and compare the delay. The number of PUs varies from 10 to 25. The number of nodes is set as 25, and the number of total channels is 10. The results show that MWIS takes the least delay, and the greedy algorithm takes the most delay. The delay of all three algorithms increase as the number of PUs in the network increases. This is because, the more PUs there are, more interference there would be.

Overall, MWIS has the best performance of average delay among the three algorithms, while the greedy algorithm takes the largest delay. The MWIS achieves almost 30% less than the greedy algorithm in delay. Moreover, the three algorithms have a similar trend when varying the network parameters.

VI. CONCLUSION

We consider the FN set selection problem in CRNs under the opportunistic routing. Three algorithms are proposed, based on the weight definition considering the channel dynamics of CRNs. One is the basic algorithm. Another one is the greedy algorithm with one backtrack scheme. The third one is the MWIS algorithm, which is the maximum weighted independent set algorithm. The simulation results show desirable performance of our algorithms.

REFERENCES

[1] I. Akyildiz, W. Lee, M. Vuran, and S. Mohanty, “Next generation/dynamic spectrum access/cognitive radio wireless networks: A survey,” *Computer Networks*, 2006.

[2] S. Biswas and R. Morris, “Exor: Opportunistic multi-hop routing for wireless networks,” in *Proc. of ACM Sigcomm*, 2005.

[3] S. Chachulski, M. Jennings, S. Katti, and D. Katabi, “Trading structure for randomness in wireless opportunistic routing,” in *Proc. of ACM Sigcomm*, 2007.

[4] E. Rozner, J. Seshadri, Y. Mehta, and L. Qiu, “Soar: Simple opportunistic adaptive routing protocol for wireless mesh networks,” *IEEE Transactions on Mobile Computing*, 2009.

[5] D. D. Couto, D. Aguayo, J. Bicket, and R. Morris, “A high-throughput path metric for multi-hop wireless routing,” in *Proc. of ACM/IEEE MobiCom*, 2003.

[6] U. Lee, P. Wang, Y. Noh, F. Vieira, M. Gerla, and J.-H. Cui, “Pressure routing for underwater sensor networks,” in *Proc. of IEEE Infocom*, 2010.

[7] Z. Wang, Y. Chen, and C. Li, “Corman: A novel cooperative opportunistic routing scheme in mobile ad hoc networks,” *IEEE Journal on Selected Areas in Communications*, 2012.

[8] N. Zhi, L. Jing, L. Botong, L. Hanchun, and X. Youyun, “A relay node selection technique for opportunistic routing in mobile ad hoc networks,” in *15th Asia-Pacific Conference on Communications*, 2009.

[9] F. Baccelli, B. Blaszczyszyn, E. Ermel, and P. Muhlethaler, “An optimized relay self selection technique for opportunistic routing in mobile ad hoc networks,” in *European Wireless Conference (EW)*, 2008.

[10] Y. Dai, J. Wu, and C. Xin, “Virtual backbone construction for cognitive radio networks without common control channel,” in *Proc. of IEEE Infocom*, 2013.

[11] J. Wu, Y. Dai, and Y. Zhao, “Local channel assignments in cognitive radio networks,” in *Proc of IEEE ICCCN*, 2011.

[12] R. Draves, J. Padhye, and B. Zill, “Comparison of routing metrics for static multi-hop wireless networks,” in *Proc. of ACM Sigcomm*, 2004.

[13] V. Lozin and M. Milanic, “A polynomial algorithm to find an independent set of maximum weight in a fork-free graph,” *Journal of Discrete Algorithms*, 2008.