Chapter 9
Routing Schemes for Cognitive Radio Mobile Ad Hoc Networks

Jun Li, Yifeng Zhou, and Louise Lamont

Abstract In this chapter, we propose a classification of existing routing schemes for cognitive radio mobile ad hoc networks (CR-MANETs) and review these representative CR-MANET routing schemes. Then, we describe a CR-MANET model and present a novel adaptive routing design for the CR-MANET, referred to as ARDC, algorithmically and through examples. ARDC is based on the graph modeling approach, and its most significant contribution is that ARDC adapts to dynamic changes in the network topology much more computationally efficient than other CR-MANET routing schemes. At last, some further research directions on CR-MANET routing are identified.

9.1 Introduction

Wireless services have been witnessing a phenomenal growth since mid-1990s. Under the current static spectrum assignment policy, each wireless device occupies a fixed portion of the spectrum for temporally and spatially exclusive usage. Consequently, the spectrum scarcity problem will be encountered in the near future as more and more wireless services are launched. That is, the frequency spectrum available for those new wireless services will be completely drained off. Following a report by the Federal Communications Commission (FCC) that many statically allocated frequency spectrum bands are under-utilized geographically and/or temporally [9], the emerging cognitive radio technology has been proposed as a solution to the spectrum scarcity problem. In contrast to the fixed and inflexible spectrum occupancy, a cognitive radio technology enabled device, referred to as a secondary user or a cognitive user, is not assigned to any fixed block of the spectrum and can operate on any vacant portion of the spectrum that has previously been allocated to a licensed user, referred to as a primary user. As indicated from their names, a primary user (PU) has high priority in the usage of its pre-assigned portion of the spectrum over a cognitive user (CU), and the activity of a PU will not be interfered

J. Li (✉)
Communications Research Centre Canada, Ottawa, ON K2H 8S2, Canada
e-mail: jun.li@crc.gc.ca

by the existence of CUs. To do so, a CU must periodically sense vacant portions of the spectrum, adaptively choose transmission parameters, and dynamically access these (under-utilized) frequency channels.

Studies on cognitive radio communication and networking have mainly been focused on lower layer (i.e., physical and medium access control layer) issues, such as spectrum sensing and opportunistic spectrum access [7, 16, 17]. Due to some fundamental differences between traditional wireless networks and cognitive radio wireless networks, the networking issues also need to be addressed [2, 3], which appear more important in a multi-hop infrastructure-less cognitive radio network, for example, a mobile ad hoc network composed of geographically colocated PUs and CUs (CR-MANET). (A cognitive radio network can also be deployed as an infrastructure-based network [2].) Due to a variety of applications of ad hoc networks in the civilian and military domain [15], in this chapter we focus on the network layer issues in self-organizing cognitive radio networks or CR-MANETs. Specifically, we concentrate on existing and new routing schemes for CR-MANETs that perform computation of end-to-end routing paths joint with channel assignment and maintenance of these routes.

In a CR-MANET, each CU is equipped with one or more pre-defined radios (or transceivers) that can be tuned to a radio frequency band (or a channel) among a range of the spectrum. In addition, a CU has the functionality of scanning available channels at the present moment to avoid inference with the activity of a PU. Through the periodic exchange of beacon information, a CU discovers its neighboring CUs, each of which connects to the CU via one or more scanned channels. A scanned channel is assumed to be symmetric in this chapter. Moreover, the maximum number of channels that can be sensed by a CU is limited. The CR-MANET is heterogenous; i.e., the set of available channels and the number of tunable transceivers may vary from one node to another. In general, the number of pre-defined transceiver is smaller than the maximum number of channels that a node can sense. Links in the CR-MANET may change over time. That is, as the network evolves over time, a new channel may become available to connect a pair of CUs, while an existing channel may disappear in the network, e.g., due to node mobility or start of occupancy of PUs. Based on the network settings discussed above, a variety of routing schemes have been proposed in the literature. In this chapter, we classify those routing schemes for CR-MANETs and review one or more representative ones for each routing class. Then, we present a novel framework of adaptive routing design for CR-MANETs, and give some concrete examples for demonstration of the adaptiveness and performance of the adaptive routing design framework. Some challenges and trends in CR-MANET routing design are also discussed.

The rest of this chapter is organized as follows: Section 9.2 classifies existing CR-MANET routing schemes and reviews these representative ones in each routing class. A novel adaptive routing design for CR-MANETs is presented and concrete examples are given in Section 9.3. Concluding remarks are given in Section 9.4.

9.2 CR-MANET Routing Schemes

In this section, we present a classification of existing routing schemes for CR-MANETs, and in each routing class, we review one or more representative schemes.

9.2.1 Classification of CR-MANET Routing Schemes

Several classification methods have been proposed to categorize existing routing schemes for CR-MANETs. In [2], CR-MANET routing schemes are categorized based on whether a scheme considers support for single or joint functionality among spectrum decision, PU awareness, and reconfigurability. In [4], the authors classify existing CR-MANET routing schemes based on whether or not the spectrum knowledge is fully captured by a CU. In this chapter, we shall present a new classification of CR-MANET routing schemes based on the approaches used for establishing end-to-end routes. As illustrated in Fig. 9.1, at the top level are two general classes. One class includes CR-MANET routing schemes designed by modifying classical MANET routing protocols, while the other contains CR-MANET routing schemes proposed by using various modeling methods. The latter has three subclasses at the second level.

9.2.2 MANET Protocol-Based CR-MANET Routing

In the class of MANET protocol-based CR-MANET routing, a routing scheme for CR-MANETs has been designed by modifying a classical MANET routing scheme. In the following, we mainly focus on SEARCH in [6] and the routing and spectrum assignment protocol in [28], and briefly discuss some other CR-MANET routing schemes, such as these in [5, 10, 18, 21, 26, 30], belonging to this routing class.
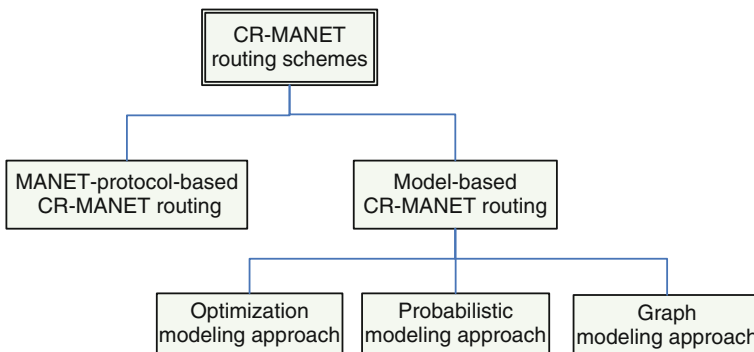

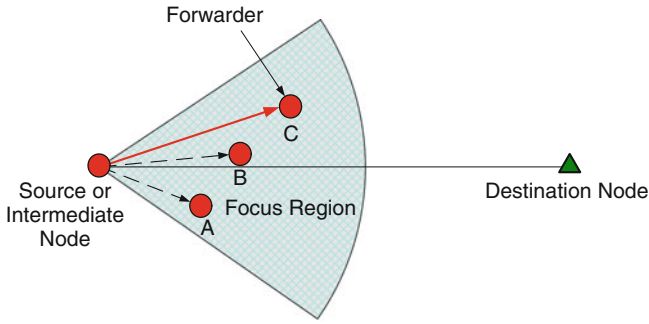**Fig. 9.1** Classification of CR-MANET routing schemes

Fig. 9.2 Greeding geographic forwarding mechanism

In [6], Chowdhury and Felice propose a routing protocol for CR-MANETs, referred to as SEARCH, based on GPSR [12], a geographic routing algorithm for MANETs. SEARCH helps a CU find the route of minimum end-to-end latency to another CU with considerations of spectrum selection and avoidance of PU activities. It is composed of two phases: route setup and route maintenance. In the route setup phase, SEARCH uses the greedy geographic forwarding mechanism, as shown in Fig. 9.2, to forward the route request (RREQ), if one or more candidate forwarders are found (i.e., some nodes are within the focus region but not covered by the area of the PU activity). Otherwise, the PU avoidance mechanism, illustrated in Fig. 9.3, is applied to find a node outside the focus region to forward RREQ. Once RREQ reaches the destination, the destination node selects a routing path together with the channels along the path using the joint channel-path optimization mechanism, which aims at minimizing the end-to-end latency. As illustrated in Fig. 9.4, a switch from one channel to another could happen on a selected route, and some extra delay induced by the channel switch has to be considered. After a routing path between the source and the destination node is set up, it might be unusable due
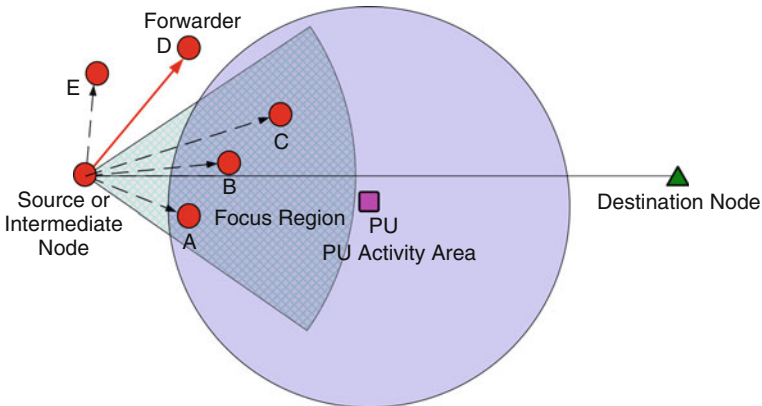
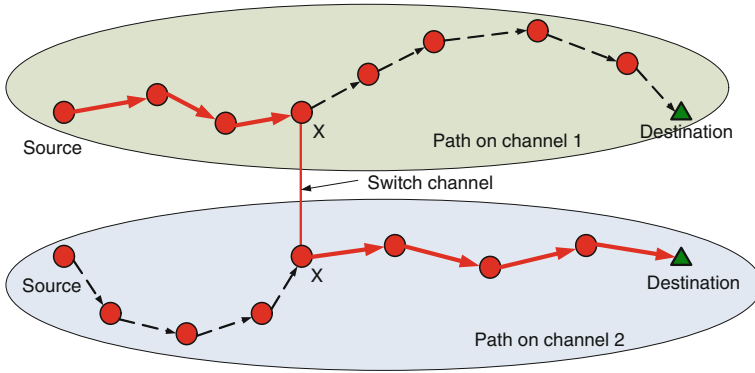

Fig. 9.3 PU avoidance mechanism

Fig. 9.4 Joint channel-path optimization mechanism

to interference with the PU activity or be disconnected because of node mobility. In either case, the route needs to be maintained in the route maintenance phase, where the node at breaking point of the route sends RREQ to the destination for a path as a replacement of the broken segment of the route. Due to the property of on-demand route setup, SEARCH runs in a decentralized manner. However, it has to be assumed that each node has the location information about its neighboring nodes and the source node knows the location of itself and the destination node. This requires some location service beneath SEARCH. Moreover, RREQ has to be transmitted along each cognitive channel in the route setup phase. All these could result in a significant amount of protocol overhead.

In [28], the authors propose an AODV-based routing and spectrum assignment protocol with local coordination of traffic flows. The protocol operates on a common control channel shared by all CUs. When an end-to-end path between a source–destination pair needs to be established, the source node broadcasts RREQ, which contains the information about its current available cognitive channels, over the common control channel. The forwarding rule for RREQ is discussed as follows. Assume that CU A broadcasts RREQ after updating the channel information in RREQ with the set S_A of available cognitive channels in A and CU B is a neighbor of A. After receiving RREQ, B determines the routing path and sends RREP to the source if B is the destination. Otherwise, the intersection of S_A and S_B is calculated. If the intersection of the two sets is an empty set, it is implied that a route passing through B cannot be established. Otherwise, B broadcasts RREQ after updating its channel information with S_B (i.e., the set of available cognitive channels in B). The procedure of processing RREQ discussed above is illustrated in Fig. 9.5. Under the same network setup as that in [28], e.g., a common control channel is shared by CUs, the authors in [21] study independently the routing problem in CR-MANETs and propose SPEAR, an AODV-based routing scheme as well.

In contrast to the CR-MANET routing schemes in [21, 28], the schemes proposed in [10, 18, 26, 30] do not count on a dedicated common control channel. Instead, RREQ is required to be broadcasted over each cognitive channel such that multiple routes, one per cognitive channel, between a source–destination pair
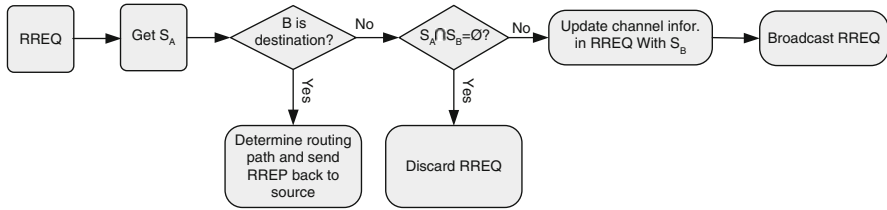
Fig. 9.5 Procedure of processing RREQ

could be established in [10, 18]. (A major difference between the two AODV-based schemes is that the deafness problem defined in [22] is explicitly addressed in [18].) The spectrum-tree-based on-demand routing protocol (STOP-RP) presented in [30] is another AODV-based routing scheme for CR-MANETs. In STOP-RP, a tree, referred to as a spectrum-tree, is constructed for each cognitive channel. Then a route discovery process, which is considered as an extended version of AODV, is conducted on the formed spectrum-trees to find the route between a pair of CUs. Another similar AODV-based routing scheme is reported in [5], where whether RREQ needs to be broadcasted via a dedicated common channel channel is not explicitly described. When a CU is equipped with multiple cognitive transceivers, a DSR-based CR-MANET routing scheme is proposed in [26] to find multiple routes with minimum contention and interference among cognitive channels.

In summary, most MANET protocol-based routing schemes for CR-MANETs are based on AODV and thus are reactive in nature. A fixed common control channel is assumed in design of some MANET protocol-based routing schemes, while it is not in others. While the feasibility of allocating a dedicated control channel in cognitive radio networks is yet to be investigated, performance comparisons of these CR-MANET routing schemes with or without the support of the common control channel need to be carried out.

9.2.3 Model-Based CR-MANET Routing

In the class of model-based CR-MANET routing, a routing scheme is proposed by solving a classic mathematical model that characterizes the settings of a real network. Therefore, a model-based CR-MANET routing scheme does not directly work upon a real network, which becomes a major difference from a MANET protocol-based routing scheme. Based on the specific modeling approaches used for establishing end-to-end routes, model-based CR-MANET routing schemes are further classified into three subclasses: optimization modeling, probabilistic modeling, and graph modeling (see Fig. 9.1).

9.2.3.1 Optimization Modeling Approach

Below we review a representative routing scheme reported in [11] and briefly discuss other CR-MANET routing schemes based on optimization modeling, such as those in [8, 19, 20].
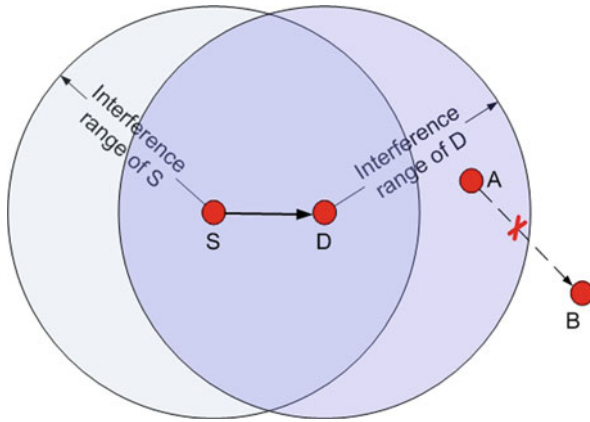
Fig. 9.6 Signal interference constraint

In [11], the authors consider a number of CUs each having a set of spectrum bands available for transmission. Each available spectrum band can be divided into a number of sub-bands of different bandwidths. The spectrum selection mechanism takes into account a signal interference model, which allows at most one communication session to use a specific sub-band in the area covered by the interference ranges of the transmitter and the receiver (see Fig. 9.6). Multiple routes between a source–destination pair can be obtained for a communication session, even though the total rate of the multiple routes needs to be equal to the required data rate of the communication session. Give a set of communication sessions (i.e., a set of source–destination pairs) and their required data rates, a non-linear optimization problem is formulated to find the routes and spectrum sub-bands passed by the routes for each communication session. The object function of the optimization problem is the total spectrum resource in the network fulfilling the communication sessions and their required rates. In contrast, Ma and Tasng [19, 20] consider a fairness factor of a communication session as the object function in their non-linear optimization problem formulation. In addition, the frequency bands of multiple channels are not divided into sub-channels. It is worthwhile noting that, even though the modeling techniques in both studies are mathematically sound, the solutions are so difficult to obtain that only approximate or heuristic results have been developed. In [8], the authors propose a CR-MANET routing scheme by formulating an optimization problem with the objective of maximizing the network throughput.

9.2.3.2 Probabilistic Modeling Approach

The probabilistic modeling approach is used in [13, 23, 24] for CR-MANET routing design. In [13], the authors consider N CUs operating in an area over a maximum number M of frequency channels. PUs are distributed in the area according to a Poisson point process, which leads to a known result that the interference of a CU with a PU approximately follows a lognormal distribution. Based on the standard

fact (e.g., Shannon's Theorem), the capacity of a channel connecting a pair of CUs is a random variable. After defining the weight of the channel as the probability that the channel capacity is not less than a fixed required rate, a path selection and channel assignment algorithm is proposed to determine the most probable path between two CUs. The implementation of the proposed routing scheme needs a global view of the network topology. In [23, 24], the authors argue that, if a channel was reliable before the present, it is more likely to be reliable in the future. Based on this argument, the weight of a channel is defined as a function of the history of temporal usage of the channel, and probabilistic routing schemes are proposed.

9.2.3.3 Graph Modeling Approach

The studies in [1, 14, 25, 27, 29] propose CR-MANET routing schemes based on the graph modeling approach. In the following, we mainly discuss the schemes reported in [1, 25, 27, 29].

In [1], Gymkhana, which is a connectivity-based routing scheme for CR-MANETs, is proposed using the graph modeling approach. In Gymkhana, the destination CU first collects the information about all possible paths through RREQs initiated by the source CU. A graph is constructed for each path at the destination, from which the Laplacian matrix of the graph is obtained. (The Laplacian matrix of a graph is defined as the difference between the graph's degree matrix and its adjacent matrix.) By using a known result that the connectivity of a graph can be measured by the second smallest eigenvalue of the Laplacian matrix of the graph, the destination CU selects the path with the highest connectivity. That is, a selected path is guaranteed to have the least interference with primary users. It is worthwhile noting that Gymkhana operates in a distributed manner, while some theoretical aspects of this scheme need be further investigated.

In [27], the authors propose a CR-MANET routing scheme based on graph modeling. The proposed scheme contains two algorithmic components: topology formation algorithm, which constructs a layered graph, and path-centric channel assignment algorithm, which conducts path computation joint with channel assignment. The number of layers in the layered graph corresponds to the maximum number of channels that can be sensed by a node. Each layer contains a set of vertices that is twice the network size (i.e., the number of CUs in the network). With four types of weighted edges connecting vertices, an optimal routing path, which has the smallest number of hops among the paths with minimum adjacent hop interference, is obtained for a pair of nodes. Since the resulting layered graph model contains a larger number of vertices even for a relatively small network size, the overall computational cost of the routing scheme is high. In [29], the authors form a colored multigraph model by using different colors for distinct channels in the network. By applying a novel shortest path algorithm to the colored multigraph model, a locally optimal routing path can be obtained for a pair of nodes. Using graph modeling algorithms, the authors in [25] propose a route selection mechanism that maximizes throughput and minimizes channel interference.

In CR-MANETs, the number of channels that are available to a CU is time varying, i.e., the available channels in the CU change as time evolves. This requires that the routing should explicitly consider the time-varying nature of the channels in an adaptive manner. A majority of above reviewed studies on CR-MANET routing design have assumed static single- or multi-channel networks without consideration of the time-varying availability of network links. In particular, almost all studies on CR-MANET routing using the graph modeling approach only address networks with static links and do not explicitly consider the time-varying nature of link availability. Another interesting aspect in CR-MANET routing design is the exploitation of the channel diversity in a CR-MANET to improve routing efficiency, since two CUs in the CR-MANET are typically connected by multiple paths through different intermediate CUs and channels. In the next section, we propose a framework of adaptive routing design for CR-MANETs using the graph modeling approach. The proposed CR-MANET routing design includes a novel topology formation component and a routing scheme. The topology formation component forms a weighted directional graph model for a given CR-MANET and adapts it to time-varying changes of network links. Based on the graph model, the routing scheme computes optimal routing paths for a pair of CUs.

9.3 ARDC: A Graph Model-Based Routing Scheme

In this section, we describe a cognitive radio mobile ad hoc network model and detail a graph model-based routing design, denoted by ARDC, for the CR-MANET.

9.3.1 CR-MANET Model and Routing Design Framework

A cognitive radio ad hoc network consists of M nodes identified by node $1, 2, \cdots, M$. Each node has one or more pre-defined transceivers that can be tuned to a radio frequency band (or a channel) among a spectrum range using its spectrum mobility and sharing functions. The number of transceivers in node m is denoted by r_m, for $m = 1, 2, \cdots, M$, and the row vector $\mathbf{r} = (r_1, r_2, \cdots, r_M)$ is referred to as the network interface vector. A node periodically scans available channels using its spectrum sensing function and discovers its neighboring nodes, each of which connects to the node via one or more scanned channels. A scanned channel is assumed to be symmetric. The maximum number of channels that can be sensed by a node is assumed to be N, and the N channels are identified by channel $1, 2, \cdots, N$. The network is heterogenous; i.e., the set of available channels and the number of tunable transceivers may vary from one node to another. In general, the number of pre-defined transceivers is smaller than the maximum number of channels that a node can sense. Links in the cognitive radio ad hoc network change over time. That is, as the network evolves over time, a new channel may become available to connect a pair of nodes, while an existing channel may disappear in the network.

We assume that one update in the cognitive radio ad hoc network takes place at a time. Specifically, we use an increasing sequence $\{t_0, t_1, t_2, \cdots\}$ of non-negative real numbers to represent a set of time instants. t_0 denotes an initial time point at which the network starts operation. At each time t_i, for $i \geq 1$, one update in the network occurs, which will be either one of the following four events.

1. A new channel becomes available between a pair of nodes;
2. An existing channel disappears in the network;
3. A communication session is initiated;
4. A communication session is finished.

In the rest of this chapter, a channel update refers to item 1 or 2, and a communication update refers to item 3 or 4. We further assume that the availability of the channels along a chosen optimal routing path for a communication session doesn't change during the period of the session. Based on the above network model, we define that an optimal routing path between a pair of nodes has the smallest number of hops after minimizing the adjacent hop interference. As defined in [29], adjacent hop interference of a route is the number of hops along the route for which each hop uses the same channel as does its previous hop.

The design framework of ARDC, as described in Algorithm 1 using an algorithmic format, consists of two major components. One component is topology formation (i.e., initialization and adaptation steps in Algorithm 1) and the other is routing scheme. The topology formation component is responsible for channel updates as well as initial graph construction, while the routing scheme deals with communication updates. The proposed routing design framework addresses routing issues that occur in the network layer. Below the network layer is the medium access control (MAC) layer, where some MAC protocol is assumed. Since a MAC protocol can regulate the access of the network nodes to the transmission channels, in practice one channel could simultaneously be used by more than one routing path. Therefore, we allow a channel to be concurrently selected by multiple routes in ARDC.

Algorithm 1 : Framework of ARDC

begin
    Initialization according to Algorithm 2 (see Section 9.3.2.1);
    $i \longleftarrow 1$;
    while *(algorithm running)* **do**
        if *(a channel update occurs at t_i)* **then**
            Adaptation according to Algorithm 3 (see Section 9.3.2.2);
        end
        if *(a communication update occurs at t_i)* **then**
            Routing scheme according to Algorithm 4 (see Section 9.3.3);
        end
        $i \longleftarrow i + 1$;
    end
end

In the remainder of this section, we discuss these two components, algorithmically and through examples.

9.3.2 Topology Formation

In this section, the topology formation component, which includes both the initialization and the adaptation steps of Algorithm 1, is explained in detail. The key concept used in topology formation algorithms is graph modeling, through which a simple directed graph is formed as a representation of the up-to-date physical network. Based on the simple directed graph model, the routing scheme, which will be detailed in Section 9.3.3, computes optimal routing paths for a requested communication session.

We denote a simple directed graph by

$$\mathcal{G}_i = (\mathcal{V}_i, \mathcal{E}_i) \tag{9.1}$$

which will be a representation of the physical network at time t_i, for $i = 0, 1, 2, \cdots$. In (1), $\mathcal{V}_i$ and $\mathcal{E}_i$ represent the vertex set and the edge set, respectively. A directional edge $\mu\nu$ represents a directional connection from vertex μ to vertex ν, while a bidirectional edge $\mu\nu$ represents two directional edges $\mu\nu$ and $\nu\mu$. In the initialization step of Algorithm 1, an initial graph $\mathcal{G}_0$ is constructed with the network connection information at the initial time t_0 and the network interface vector $\mathbf{r}$ being inputs. If a new channel becomes available or an existing channel disappears at time t_i, for $i \geq 1$, an adaptive graph $\mathcal{G}_i$ is created by adapting $\mathcal{G}_{i-1}$ to the channel update at t_i in the adaptation step.

In the constructed graph $\mathcal{G}_i$, for each $i = 0, 1, 2, \cdots$, a vertex $m_n \in \mathcal{V}_i$ implies that node m has channel n available for communicating with a neighbor of the node at time t_i. Directional edges from/to vertex m_n are added between vertices associated with neighbors of node m. A directional edge $m_n\hat{m}_{\hat{n}}$ implies that data packets could enter node m over channel n and be routed from node m to node $\hat{m}$ over channel $\hat{n}$.

We take into account the adjacent hop interference metric and the effect of transceivers in optimal routing path computation by assigning a weight associated with each directional edge. First of all, on a selected routing path, it is preferable that two adjacent hops use different channels to minimize adjacent hop interference. This is accomplished by differentiating directional edges of $\mathcal{G}_i$. That is, the weight associated with a directional edge $m_n\hat{m}_{\hat{n}}$ for $n \neq \hat{n}$ must be smaller than that associated with edge $m_n\hat{m}_{\hat{n}}$ for $n = \hat{n}$. Moreover, a node with more than one transceivers can relay packets simultaneously using two different channels (one for reception and the other for transmission), and thus should be considered a preferable intermediate node on an optimal routing path. Therefore, the assigned weight of a directional edge $m_n\hat{m}_{\hat{n}}$ for $n \neq \hat{n}$ depends on the number r_m of transceivers in node m. In summary, if we let w be the weight associated with a directional edge $m_n\hat{m}_{\hat{n}}$ for $n = \hat{n}$, w_1 the weight associated with a directional edge $m_n\hat{m}_{\hat{n}}$ for $n \neq \hat{n}$ and $r_m = 1$, and w_2 the weight associated with a directional edge $m_n\hat{m}_{\hat{n}}$ for $n \neq \hat{n}$ and $r_m \geq 2$, then we have

$$w > w_1 > w_2 \geq 0 \tag{9.2}$$

In examples and simulation studies conducted in this Chapter, we use $w = 3$, $w_1 = 2$, and $w_2 = 1$ for illustrating the algorithm. A study on optimized settings of these weights could be carried out as a future research topic.

9.3.2.1 Initialization Step

The algorithm for constructing the initial graph $\mathcal{G}_0$ is given by Algorithm 2. In this algorithm, we create vertices associated with each node in the CR-MANET. A vertex m_n is added to the vertex set $\mathcal{V}_0$ if node m can potentially communicate with a neighbor of node m over channel n. A directional edge $m_n\hat{m}_{\hat{n}}$ indicates that data packets can be received by node m via channel n and routed from node m to node $\hat{m}$ over channel $\hat{n}$. This directional edge can result in minimum (or zero) adjacent hop interference if $n \neq \hat{n}$, and thus should be favorably chosen on an optimal routing path. Therefore, the assigned weight of directional edge $m_n\hat{m}_{\hat{n}}$ for $n \neq \hat{n}$ and $r_m > 1$ (w_2 in Algorithm 2) is smaller than that for $n \neq \hat{n}$ and $r_m = 1$ (w_1 in Algorithm 2), which is smaller than that for $n = \hat{n}$ (w in Algorithm 2).

Algorithm 2 : Initialization Step

---

input : The physical network at t_0 and the interface vector $\mathbf{r} = [r_1, r_2, \cdots, r_M]$.
output: Graph $\mathcal{G}_0$ with all edges assigned to weights.
begin
    $\mathcal{V}_0 \longleftarrow \emptyset$;
    $\mathcal{E}_0 \longleftarrow \emptyset$;
    for $m \leftarrow 1$ **to** M **do**
        temp $\leftarrow \mathcal{V}_0$;
        for $n \leftarrow 1$ **to** N **do**
            if *channel n is available at node m* **then**
                add a new vertex m_n into $\mathcal{V}_0$;
                foreach *vertex $\hat{m}_{\hat{n}}$ in* temp **do**
                    if *($n \neq \hat{n}$) & (m and $\hat{m}$ is connected via channel $\hat{n}$)* **then**
                        add directional edge $m_n\hat{m}_{\hat{n}}$ to $\mathcal{E}_0$;
                        assign weight of directional edge $m_n\hat{m}_{\hat{n}}$ to w_1 **if** $r_m = 1$;
                        assign weight of directional edge $m_n\hat{m}_{\hat{n}}$ to w_2 **if** $r_m \geq 2$;
                    end
                    if *($n \neq \hat{n}$) & (m and $\hat{m}$ is connected via channel n)* **then**
                        add directional edge $\hat{m}_{\hat{n}}m_n$ to $\mathcal{E}_0$;
                        assign weight of directional edge $\hat{m}_{\hat{n}}m_n$ to w_1 **if** $r_{\hat{m}} = 1$;
                        assign weight of directional edge $\hat{m}_{\hat{n}}m_n$ to w_2 **if** $r_{\hat{m}} \geq 2$;
                    end
                    if *($n = \hat{n}$) & (m and $\hat{m}$ is connected via channel n)* **then**
                        add bidirectional edge $m_n\hat{m}_n$ to $\mathcal{E}_0$;
                        assign weight of directional edge $m_n\hat{m}_n$ to w;
                    end
                end
            end
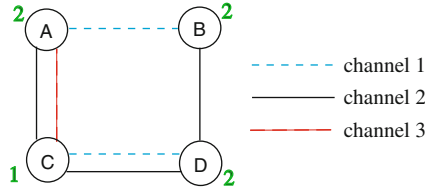   **end**
    end
end

Fig. 9.7 Cognitive radio ad hoc network

In the following, Example 1 illustrates Algorithm 2, which creates the initial graph $\mathcal{G}_0$.

Example 1 Assume that a cognitive radio ad hoc network consists of nodes identified by A, B, C, D ($M = 4$). The maximum number of channels that can be sensed by a node is 3 ($N = 3$), and these channels are identified by channel 1, 2, and 3. The physical network as shown in Fig. 9.7, where the integer near a node represents the number of transceivers of the node, is given at some initial time t_0.

Initially, we set two empty sets $\mathcal{V}_0$ and $\mathcal{E}_0$. Since $M = 4$, we need four iterations, each of which corresponds to one of these four nodes, to complete construction of graph $\mathcal{G}_0$. Without loss of generality, we use the alphabetic order of the nodes for the construction process. In the first iteration, we consider node A, whose available channels are channels 1, 2, and 3. After this iteration, $\mathcal{V}_0 = \{A_1, A_2, A_3\}$ and $\mathcal{E}_0$ is empty. $\mathcal{G}_0$ after the first iteration is shown in Fig. 9.8a. The second iteration involves node B, which has channels 1 and 2 available and is connected with node A over channel 1. Then two vertices B_1 and B_2 are added to the set $\mathcal{V}_0$. Since $r_A = r_B = 2$, these added directional edges have weight 1 except bidirectional edge $A_1 B_1$ with weight 3. Graph $\mathcal{G}_0$ after this iteration is shown in Fig. 9.8b. Node C is dealt with in the third iteration. It has three available channels, and thus three new vertices C_1, C_2, and C_3, are added into the set $\mathcal{V}_0$. Since $r_C = 1$, these directional edges $C_n A_{\hat{n}}$, for which $n \neq \hat{n}$, have weight 2. $\mathcal{G}_0$ after the third iteration is shown in Fig. 9.9.
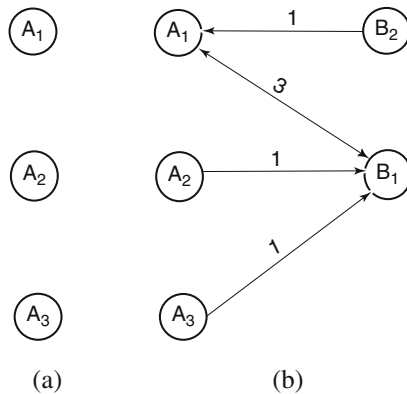

(a) (b)

Fig. 9.8 First and second iterations

Fig. 9.9 Third iteration

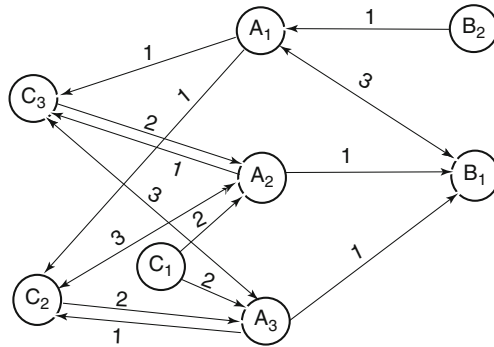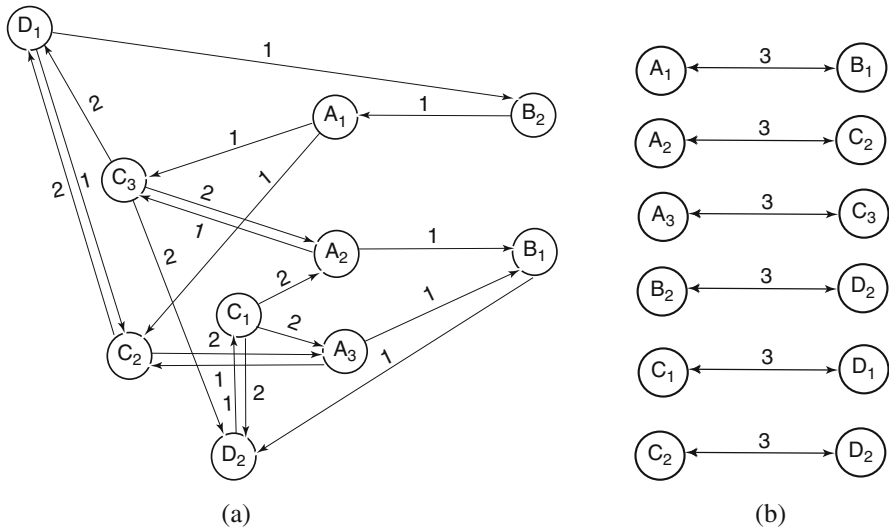(a)   (b)

Fig. 9.10 Graph $\mathcal{G}_0$. (**a**) Edges of weights 1 and 2; (**b**) Edges of weight 3

In the last iteration, two vertices D_1 and D_2 are added to the set $\mathcal{V}_0$ followed by corresponding directional edges. Now the vertex set $\mathcal{V}_0$ is complete and the construction process is terminated. The final graph $\mathcal{G}_0$ is shown in Fig. 9.10, where edges of weights 1 and 2 are shown in Fig. 9.10a and edges of weight 3 are shown in Fig. 9.10b.

9.3.2.2 Adaptation Step

A channel update is caused by either the availability of a new channel or the disappearance of an existing channel from the network. Based on $\mathcal{G}_{i-1}$ and the channel update at time t_i, an adaptive algorithm for constructing $\mathcal{G}_i$ is given in Algorithm 3. As we can see from the algorithm, a channel update only affects the topology
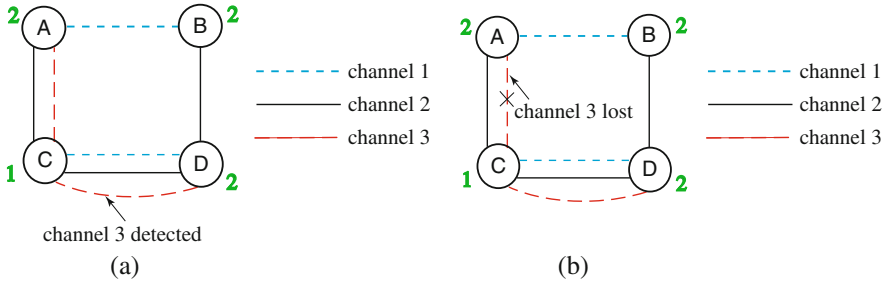
Fig. 9.11 Channel updates. (**a**) Gain of a channel; (**b**) Loss of a channel

formation locally. Thus we can locally adapt $\mathcal{G}_{i-1}$ with the channel update. This results in the adaptive graph $\mathcal{G}_i$, which represents the topology formation of the physical network at t_i. Algorithm 3 in the adaptation step features a very low computational cost, which is promising for practical implementations in CR-MANETs. The following example illustrates Algorithm 3 with $\mathcal{G}_{i-1}$ and a channel update at t_i being inputs.

Example 2 At time t_0, the physical network as shown in Fig. 9.7 with corresponding initial graph $\mathcal{G}_0$ formed in Fig. 9.10 is given. At some time t_1 after t_0, a channel update (i.e., gain of channel 3 connecting C and D) takes place, as shown in Fig. 9.11a. This results in $\mathcal{G}_1$. At some time t_2 after t_1, another channel update (i.e., loss of channel 3 connecting A and C before t_2) occurs, as shown in Fig. 9.11b. This results in $\mathcal{G}_2$. We construct adaptive graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ below based on Algorithm 3.

To obtain adaptive graph $\mathcal{G}_1$, graph $\mathcal{G}_0$, and the channel update at time t_1 (i.e., channel 3 between C and D) are used as inputs to Algorithm 3. Initially, a new vertex D_3 is added to the vertex set $\mathcal{V}_1$. (Vertex C_3 is already in $\mathcal{V}_1$.) Then, directional edges are added to the edge set $\mathcal{E}_1$. The resulting graph $\mathcal{G}_1$ after adapting $\mathcal{G}_0$ to the acquisition of channel 3 at t_1 is shown in Fig. 9.12, where the newly added vertex and edges are shown in dotted lines.

At time t_2, channel 3, which connected A and C before t_2, becomes unavailable. Graph $\mathcal{G}_1$ and loss of channel 3 are used as inputs to Algorithm 3. In order for $\mathcal{G}_1$ to adapt to the update, directional edges A_1C_3, A_2C_3, A_3C_3, C_1A_3, C_2A_3, and C_3A_3 are removed from graph $\mathcal{G}_1$. Since channel 3 becomes unavailable in A at time t_2, vertex A_3 and all corresponding edges involving A_3 are removed from $\mathcal{G}_1$. The resulting graph $\mathcal{G}_{t_2}$ after adapting to the loss of channel 3 at t_2 is shown in Fig. 9.13, where the removed vertex and edges are in dashed lines.

9.3.3 Routing Scheme

The routing scheme computes optimal routing paths for a requested communication session and updates the topology formation for a completed communication session.

Algorithm 3: Adaptation Step

---

input : Graph $\mathcal{G}_{i-1}$ and channel update at t_i.
output: Graph $\mathcal{G}_i$ with all edges assigned to weights.
begin

    $\mathcal{V}_i \longleftarrow \mathcal{V}_{i-1}$, and $\mathcal{E}_i \longleftarrow \mathcal{E}_{i-1}$;
    $boolean1 \longleftarrow False$, and $boolean2 \longleftarrow False$;
    if *channel k connecting node m and $\hat{m}$ is GAINED at t_i* **then**
        if $m_k \notin \mathcal{V}_i$ **then** add new vertex m_k into $\mathcal{V}_i$, and $boolean1 \longleftarrow True$;
        if $\hat{m}_k \notin \mathcal{V}_i$ **then** add new vertex $\hat{m}_k$ into $\mathcal{V}_i$, and $boolean2 \longleftarrow True$;
        for $n \leftarrow 1$ **to** N **do**
            if *(n $\neq$ k) & (m_n is in $\mathcal{V}_i$)* **then**
                add directional edge $m_n\hat{m}_k$ to $\mathcal{E}_i$;
                assign weight of directional edge $m_n\hat{m}_k$ to w_1 **if** $r_m = 1$ OR to w_2 **if**
                $r_m \geq 2$;
            end
            if *(n $\neq$ k) & ($\hat{m}_n$ is in $\mathcal{V}_i$)* **then**
                add directional edge $\hat{m}_n m_k$ to $\mathcal{E}_i$;
                assign weight of directional edge $\hat{m}_n m_k$ to w_1 **if** $r_{\hat{m}} = 1$ OR to w_2 **if**
                $r_{\hat{m}} \geq 2$;
            end
            if *(n $\neq$ k) & (m and $\hat{m}$ is connected via channel n)* **then**
                if *(boolean1 is True)* **then**
                    add directional edge $m_k\hat{m}_n$ $\mathcal{E}_i$;
                    assign the weight of directional edge $m_k\hat{m}_n$ to w_1 **if** $r_m = 1$ OR to
                    w_2 **if** $r_m \geq 2$;
                end
                if *(boolean2 is True)* **then**
                    add directional edges $\hat{m}_k m_n$ to $\mathcal{E}_i$;
                    assign the weight of directional edge $\hat{m}_k m_n$ to w_1 **if** $r_{\hat{m}} = 1$ OR to
                    w_2 **if** $r_{\hat{m}} \geq 2$;
                end
            end
            if *(n is equal to k)* **then**
                add bidirectional edge $m_n\hat{m}_k$ into $\mathcal{E}_i$;
                assign the weight of bidirectional edge $m_k\hat{m}_k$ to w;
            end
        end
    end
    if *channel k connecting node m and $\hat{m}$ is LOST at t_i* **then**
        for $n \leftarrow 1$ **to** N **do**
            if *m_n (respectively, $\hat{m}_n$) is in $\mathcal{V}_i$* **then**
                remove directional edge $m_n\hat{m}_k$ (respectively, $\hat{m}_n m_k$) from $\mathcal{E}_i$;
            end
        end
        if *channel k is not available in node m (respectively, node $\hat{m}$) at t_i* **then**
            remove all edges involving m_k (respectively, those involving $\hat{m}_k$) from $\mathcal{E}_i$;
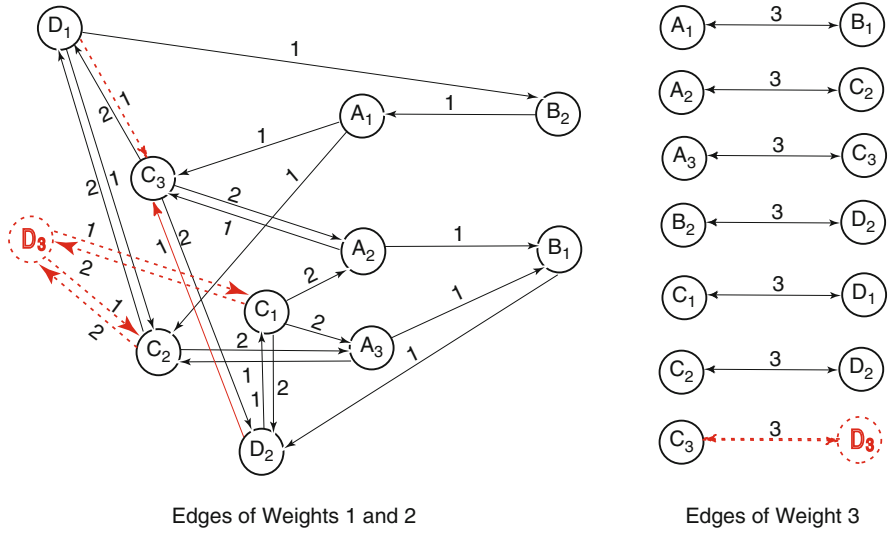   remove vertex m_k (respectively, vertex $\hat{m}_k$) from $\mathcal{V}_i$;
        end
    end
end

Edges of Weights 1 and 2   Edges of Weight 3

Fig. 9.12 Adaptive graph $\mathcal{G}_1$

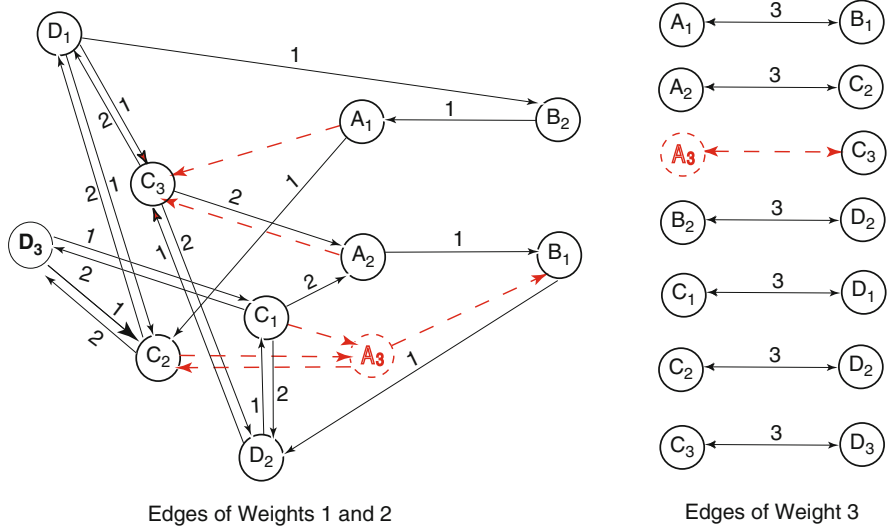Edges of Weights 1 and 2   Edges of Weight 3

Fig. 9.13 Adaptive graph $\mathcal{G}_2$

We assume that, at time t_i for some $i \geq 1$, a communication update from source node S to destination node D is made. When the communication session from node S to D is initiated at t_i, the set of available channels in S is assumed to be $\{s_1, \cdots, s_P\}$ and the set of available channels in D is $\{d_1, \cdots, d_Q\}$, for some $P, Q = 1, \cdots, N$. In this case, the routing algorithm, as described in Algorithm 4,

is used to identify an optimal routing path for the communication session and to update the topology formation.

In Algorithm 4, vertices S and D, and some directional edges involving S and D are added to assist in computing the optimal routing paths from S to D. By applying a known shortest path algorithm (e.g., Dijkstra's algorithm) to $\mathcal{G}_i$, we obtain the set $\mathcal{P}$ of all optimal routing paths from S to D. An optimal routing path $S \rightarrow S_x \rightarrow m_n \rightarrow \hat{m}_{\hat{n}} \rightarrow D_y \rightarrow D$, for some $x = s_1, \cdots, s_P$, intermediate vertices m_n and $\hat{m}_{\hat{n}}$, and $y = d_1, \cdots, d_Q$, implies that packets will be sent by node S to m over channel n, relayed from m to $\hat{m}$ via channel $\hat{n}$, relayed from $\hat{m}$ to D over channel y, and finally received by D over channel y. A study will be conducted in our future work to examine if a computed optimal routing path could contain a loop.

After the set $\mathcal{P}$ is computed, we simply select the first path $P(S, D)$ in set $\mathcal{P}$ to be used for the communication session. As discussed at the beginning of this section, a channel is allowed to be used by multiple communication sessions via a MAC protocol below the network layer. In such a case, a packet could experience a longer period of waiting time before transmission. To mitigate this effect, we adjust

---

Algorithm 4 : Routing Scheme

input : Graph $\mathcal{G}_{i-1}$ and the communication update at t_i.
output: An optimal routing path (only for the case where a communication session is initiated) and adaptive graph $\mathcal{G}_i$.
begin
    $\mathcal{V}_i \longleftarrow \mathcal{V}_{i-1}$, and $\mathcal{E}_i \longleftarrow \mathcal{E}_{i-1}$ to get $\mathcal{G}_i$;
    if *(communication session from node S to D is INITIATED)* **then**
        add two vertices S and D to $\mathcal{V}_i$;
        add directional edge SS_x to $\mathcal{E}_i$, for each $x = s_1, \cdots, s_P$, and assign its weight to zero;
        add directional edge D_yD to $\mathcal{E}_i$, for each $y = d_1, \cdots, d_Q$, and assign its weight to zero;
        find set $\mathcal{P}$ of shortest paths from S to D by applying a shortest path algorithm (*e.g.*, Dijkstra's algorithm) to $\mathcal{G}_i$;
        if *(set $\mathcal{P}$ is NOT empty)* **then**
            assign the first path $P(S, D)$ in $\mathcal{P}$ to be the optimal routing path for communication session from S to D;
            increase by 1 the weight of each directional edge along the selected optimal routing path $P(S, D)$ in $\mathcal{G}_i$;
        end
        remove directional edges SS_x and D_yD, for each $x = s_1, \cdots, s_P$ and $y = d_1, \cdots, d_Q$, from $\mathcal{E}_i$;
        remove vertices S and D from $\mathcal{V}_i$;
        return $P(S, D)$ and graph $\mathcal{G}_i$;
    end
    if *(communication session from node S to D is COMPLETED)* **then**
        in $\mathcal{G}_i$, decrease by 1 the weight of each directional edge along the optimal routing path used for the communication session from node S to D;
        return graph $\mathcal{G}_i$;
    end
end

the edge weights in graph $\mathcal{G}_i$ that correspond to channels that are being used for the communication session. This reduces the possibility of a channel being used by two simultaneous communication sessions. The adjustment process should reflect these explicit MAC requirements; this adjustment process in Algorithm 4 is implemented by increasing the weight of each directional edge along the selected optimal routing path $P(S, D)$ by 1. Once the communication session is complete, a reverse operation is performed to update the graph. That is, the weight of each directional edge along the selected optimal routing path $P(S, D)$ is decreased by 1.

The routing path computation in Algorithm 4 provides at least one optimal routing path to used for packet transmission from S to D, as long as they are connected to each other. Since the computed set $\mathcal{P}$ may consist of multiple elements, multiple routing paths from S to D can be obtained, each of which has the same number of hops from S to D with the same minimized adjacent hop interference. In the following, an example is given to illustrate how to compute optimal routing paths and obtain graph $\mathcal{G}_i$ based on Algorithm 4.

Example 3 At time t_i, a physical network as shown in Fig. 9.14 is given, and a request for transmission from source node A to destination node D is made. From Example 2, the corresponding topology formation before t_i, which is denoted by
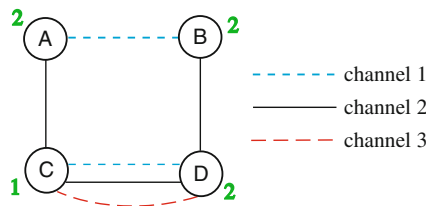

Fig. 9.14 Example 3

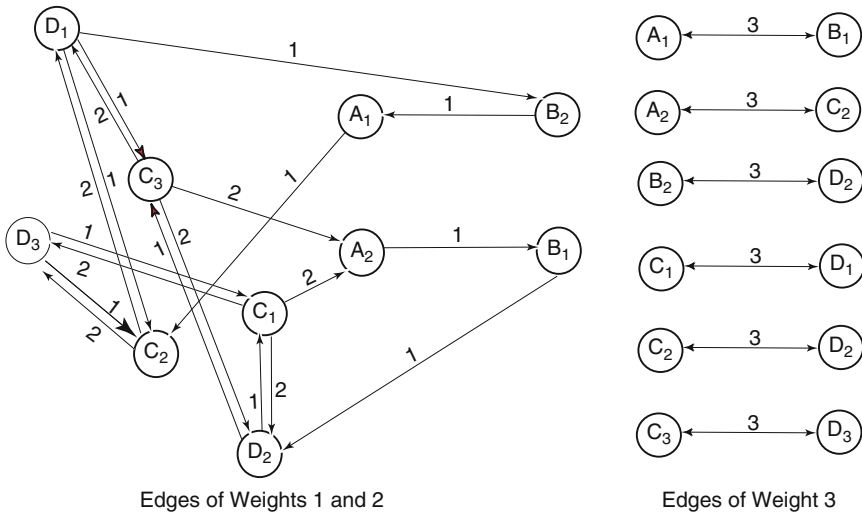Edges of Weights 1 and 2   Edges of Weight 3

Fig. 9.15 Graph model

$\mathcal{G}_{i-1}$, is given in Fig. 9.15. We need to identify the optimal routing path(s) for transmission from A to D and the updated graph $\mathcal{G}_i$ based on Algorithm 4.

Starting from graph $\mathcal{G}_{i-1}$, two vertices A and D are first added to the graph. Then directional edges AA_1, AA_2, D_1D, D_2D, and D_3D are added with weight 0. After running Dijkstra's algorithm on the graph from A to D, $\mathcal{P} = \{A \rightarrow A_2 \rightarrow B_1 \rightarrow D_2 \rightarrow D\}$ gives a unique optimal routing path $P(A, D)$. The unique transmission route means that packets shall be transmitted from A (the source node) to B through channel 1 and from B to D (the destination node) via channel 2. During this transmission process, the weights of both directional edge A_2B_1 and directional edge B_1D_2 are increased by 1, respectively, which results in graph $\mathcal{G}_i$.

9.4 Conclusion and Discussion

In this chapter, we categorized existing routing schemes for CR-MANETs based on the approaches used for solving the routing problem. For each routing class, we reviewed one or more representative routing schemes. At last, we presented ARDC, a graph model-based routing design for CR-MANETs. In contrast to most relevant studies that do not address the time-varying nature of the availability of links in CR-MANETs, ARDC is capable of adapting to dynamic changes in the network topology with little computational effort. Based on the up-to-date network topology, ARDC computes optimal routing paths for a given pair of source and destination nodes.

From review of existing routing schemes for CR-MANETs, we strongly believe that further research needs to be carried out to identify pros and cons for these routing schemes. For instance, a number of proposed CR-MANET routing schemes have to be operated in a centralized unit, while others operate in a distributed manner. It needs to be investigated which routing structure, centralized or distributed, is more appropriate for CR-MANETs. A number of schemes assumes a dedicated common control channel shared by all CUs, while others do not count on it. The feasibility of CUs equipped with a transceiver operating on a fixed common control channel needs to be studied, and performance comparisons of schemes relying or not relying on the common control channel need to be conducted. We are also convinced that significant contributions are needed to break through some major challenges in CR-MANET routing. For instance, it has been well known that the topology of a CR-MANET changes more frequently than other types of mobile networks. This necessitates novel routing schemes for CR-MANETs that are capable of more promptly and efficiently adapting to dynamic changes in the network topology. Existing CR-MANET routing schemes mainly address the route and channel selection problem for two cognitive users. Since CUs and PUs are geographically colocated, interactions and communications between a cognitive user and a primary user will take place. This requires integrating CR-MANET routing schemes that support both CU-CU and CU-PU communications.

References

1. A. Abbagnale and F. Cuomo, "Gymkhana: a connectivity-based routing scheme for cognitive radio ad hoc networks," *Proc. 29th IEEE International Conference on Computer Communications (INFOCOM 2010)*, pp. 1–5, San Diego, CA, Mar. 2010.
2. I.F. Akyildiz, W.Y. Lee, M.C. Vuran, and S. Mohanty, "NeXt generation/dynamic spectrum access/cognitive radio wireless networks: A survey," *Elsevier Computer Networks*, vol. 50, no. 13, pp. 2127–2159, Sept. 2006.
3. I.F. Akyildiz, W.Y. Lee, M.C. Vuran, and K.R. Chowdhury, "CRAHNs: Cognitive radio *ad hoc* networks," *Elsevier Ad Hoc Networks*, vol. 7, no. 5, pp. 810–836, Jul. 2009.
4. M. Cesana, C. Francesca, and E. Eylem, "Routing in cognitive radio networks: Challenges and solutions," *Elsevier Ad Hoc Networks*, vol. 9, no. 3, pp. 228–248, May 2011.
5. G. Cheng, W. Liu, Y. Li, and W. Cheng, "Joint on-demand routing and spectrum assignment in cognitive radio networks," *Proc. IEEE International Conference on Communications (ICC 2007)*, pp. 6499–6503, Scotland, Jun. 2007.
6. K.R. Chowdhury and M.D. Felice, "SEARCH: A routing protocol for mobile cognitive radio ad-hoc networks," *Elsevier Computer Communications*, vol. 32, no. 18, pp. 1983–1997, Dec. 2009.
7. C. Cormio and K.R. Chowdhury, "A survey on MAC protocols for cognitive radio networks," *Elsevier Ad Hoc Networks*, vol. 7, no. 7, pp. 1315–1329, Sept. 2009.
8. L. Ding, T. Melodia, S.N. Batalama, and M.J. Medley, "ROSA: distributed joint routing and dynamic spectrum allocation in cognitive radio *ad hoc* networks," *Proc. the 12th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM 2009)*, pp. 13–20, Spain, Oct. 2009.
9. FCC, "Spectrum policy task force report," *ET Docket No. 02 - 135*, 2002.
10. T. Fujii and Y. Yamao, "Multi-band routing for ad-hoc cognitive radio networks," *Proc. SDR'06*, Orlando, FL, Nov. 2006.
11. Y.T. Hou, Y. Shi, and H.D. Sherali, "Spectrum sharing for multi-hop networking with cognitive radios," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 1, pp. 146–154, Jan. 2008.
12. B. Karp and H.T. Kung, "GPSR: greedy perimeter stateless routing for wireless networks," *Proc. of 6th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom 2000)*, pp. 243–254, Boston, MA, Aug. 2000.
13. H. Khalife, S. Ahuja, N. Malouch, and M. Krunz, "Probabilistic path selection in opportunistic cognitive radio networks," *Proc. IEEE Global Communications Conference (GLOBECOM 2008)*, pp. 4861–4865, New Orleans, LA, Nov. 2008.
14. Y.R. Kondareddy and P. Agrawal, "A graph based routing algorithm for multi-hop cognitive radio networks," *Proc. the 4th Annual International Conference on Wireless Internet (WICON'08)*, Maui, Hawaii, Nov. 2008.
15. C. Kopp, *NCW101: An introduction to network centric warfare, Part 4 - Ad Hoc Networking*, Air Power Australia, 2008.
16. K.B. Letaief and W. Zhang, "Cooperative communications for cognitive radio networks," *Proc. of the IEEE*, vol. 97, no. 5, pp. 878–893, May 2009.
17. Z. Li, F.R. Yu, and M. Huang, "A distributed consensus-based cooperative spectrum sensing in cognitive radios," *IEEE Transactions on Vehicular Technolgy*, vol. 59, no. 1, pp. 383–393, Jan. 2010.
18. H. Ma, L. Zheng, X. Ma, and Y. Luo, "Spectrum aware routing for multi-hop cognitive radio networks with a single transceiver," *Proc. 3rd International ICST Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom 2008)*, Singapore, May 2008.

19. M. Ma and D.H.K. Tsanga, "Joint spectrum sharing and fair routing in cognitive radio networks," *Proc. 5th IEEE Consumer Communications and Networking Conference (CCNC 2008)*, pp. 978–982, Las Vegas, NV, Jan. 2008.
20. M. Ma and D.H.K. Tsanga, "Joint design of spectrum sharing and routing with channel heterogeneity in cognitive radio networks," *Elsevier Physical Communication*, vol. 2, no. 1–2, pp. 127–137, Mar.–Jun. 2009.
21. A. Sampath, L. Yang, L. Cao, H. Zheng, and B.Y. Zhao, "High throughput spectrum-aware routing for cognitive radio networks," *Proc. 3rd International ICST Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom 2008)*, Singapore, May 2008.
22. J. So and N.H. Vaidya, "A routing protocol for utilizing multiple channels in multi-hop wireless networks with a single transceiver," Technical Report, University of Illinois at Urbana-Champaign, Oct. 2004.
23. A.C. Talay and D.T. Altilar, "RACON: A routing protocol for mobile cognitive radio networks," *Proc. the 2009 ACM Workshop on Cognitive Radio Networks*, pp. 73–78, Beijing, China, Sept. 2009.
24. A.C. Talay and D.T. Altilar, "ROPCORN: Routing protocol for cognitive radio ad hoc networks," *Proc. International Conference on Ultra Modern Telecommunications & Workshops (ICUMT'09)*, pp. 1–6, St. Petersburg, Oct. 2009.
25. Q. Wang and H. Zheng, "Route and spectrum selection in dynamic spectrum networks," *Proc. IEEE Consumer Communications and Networking Conference (CNCC 2006)*, vol. 1, pp. 625–629, Jan. 2006.
26. X. Wang, T. Kwon, and Y. Choi, "A multipath routing and spectrum access (MRSA) framework for cognitive radio systems in multi-radio mesh networks," *Proc. 2009 ACM Workshop on Cognitive Radio Networks*, pp. 55–60, Beijing, China, Sept. 2009.
27. C. Xin, L. Ma, and C.C. Shen, "A path-centric channel assignment framework for cognitive radio wireless networks," *ACM/Springer Mobile Networks and Applications*, vol. 13, no. 5, pp. 463–476, Oct. 2008.
28. Z. Yang, G. Cheng, W. Liu, W. Yuan, and W. Cheng, "Local coordination based routing and spectrum assignment in multi-hop cognitive radio networks," *Springer Mobile Networks and Applications*, vol. 13, no. 1–1, pp. 67–81, Apr. 2008.
29. X. Zhou, L. Lin, J. Wang, and X. Zhang, "Cross-layer routing design in cognitive radio networks by colored multigraph model," *Springer Wireless Personal Communications*, vol. 49, no. 1, pp. 123–131, Jul. 2009.
30. G.M. Zhu, I.F. Akyildiz, and G.S. Kuo, "STOD-RP: A spectrum-tree based on-demand routing protocol for multi-hop cognitive radio networks," *Proc. IEEE Global Communications Conference (GLOBECOM 2008)*, pp. 1–5, New Orleans, LA, Nov. 2008.