A Stable Path Routing Protocol for Cognitive Radio Ad hoc Networks based on the Maximum Number of Common Primary User Channels

Natarajan Meghanathan
Associate Professor of Computer Science Jackson State University, Jackson, MS 39217, USA
Email: nmehanathan@jsums.edu

Abstract—We propose a novel local spectrum knowledge-based distributed stable path routing protocol for cognitive radio ad hoc networks (CRAHNs) where the unlicensed secondary users (SUs) make use of the licensed channels of the Primary Users (PUs) when the latter are not actively using the channels. We model a time-variant CRAHN of SUs with links between any two SUs if they have at least one common PU channel available for use in their neighborhood and the weight of an edge is the number of such common PU channels available for use. Referred to as the Maximum Common Primary User channel-based Routing (MCPUR) protocol, the proposed protocol prefers to choose an SU-SU source-destination (s-d) path with the largest value for the sum of the number of common PU channels available for use across each of its constituent edges. Our hypothesis is that such an s-d route is likely to exist for a longer time (and incur fewer broadcast route discoveries) as the end nodes of the constituent SU-SU edges are more likely to have at least one common available PU channel that can be used to complete the transmission and reception of data packets. Simulation results confirm our hypothesis to be true: the number of path transitions incurred with MCPUR could be at most 62% lower than that of the path transitions incurred with a minimum hop-based shortest path routing (SPR) protocol. The tradeoff is a low-moderate increase in the hop count per path (as large as 17%).

Index Terms—Path Stability, Cognitive radio ad hoc networks, Routing protocol, Primary User, Local spectrum knowledge

I. INTRODUCTION

With the prolific growth in the number of wireless devices and users, congestion in the widely used 900 MHz and 2.4 GHz ISM bands (Industrial, Scientific and Medical bands) is increasing exorbitantly [1]. To alleviate this problem, the Federal Communications Commission (FCC) has recently approved the use of unlicensed devices in the licensed bands. Consequently, a new wave of research has opened up on using cognitive radios for dynamic spectrum access [2] to maximize the efficiency of spectrum usage. A cognitive radio is a software-defined radio that can dynamically change its transmission parameters based on the channels available in the wireless spectrum of its operating environment [1]. A cognitive radio network (CRN) is thus a network of wireless devices embedded with cognitive radios that have the ability to sense the availability of channels in the neighborhood and switch from one channel to another, depending on the availability of the channels. Accordingly, CRNs typically comprise of two categories of users: Primary Users (PUs) who are the licensed users of the spectrum (channels with particular frequencies registered for use) and Secondary Users (SUs) who are not the licensed users, but use the licensed channels when they are idle and not used by the PUs. When a PU becomes active, the SUs have to either switch to another available channel (called spectrum overlay) or stay in the same spectrum (but alter the transmission power or modulation scheme so that the PU is not interfered with, referred to as spectrum underlay).

Most of the research conducted so far in the realm of cognitive radio networks focused on developing efficient schemes for spectrum sensing at the physical layer and maximizing channel usage (e.g., [3] [4]). Relatively, very limited work has been done towards developing efficient routing protocols that are tailored to suit the characteristics of cognitive radio networks and leverage the spectrum availability information of the lower layers. In this research, we target a category of cognitive radio networks called cognitive radio ad hoc networks (CRAHNs) and address the problem of minimizing the number of path transitions (change from one path to another by going through a global broadcast route discovery process) incurred while routing data packets for a source-destination (s-d) session between two SU nodes. A CRAHN is a self-organized ad hoc cognitive radio network [5] of the licensed primary users and the unlicensed secondary users formed impromptu without any centralized administrative control; all the nodes in a CRAHN are peers of each other. One or more SUs could share the licensed channel of a PU in their neighborhood without interfering the latter. However, since spectrum underlay is prone to introduce heterogeneity in the network of SUs as well as create unidirectional SU-SU links when the CRAHN is modeled as a graph, we assume the CRAHN considered in this research to be of spectrum overlay type. Accordingly, the SUs in a neighborhood consider a channel as available when the licensed PU of that channel (also in their neighborhood) is not active (i.e., turned OFF). When the PU becomes
active (i.e., turned ON), the SUs do not consider the PU’s channel to be available for use.

A PU channel is available for an SU only if both the PU and SU are within the transmission range of each other and that the PU channel is not in use by its licensed user. Due to the limited transmission range of the operating wireless devices (both the PUs and the SUs), a PU channel is most likely not available for sharing among all the SUs in the CRAHN. The set of PU channels available for the SUs need not be the same. Two SUs are said to have a link if there is at least one common PU channel available in their mutually intersecting neighborhood. For transmission, the upstream node on the path may have to tune its data channel to one of the available PU channels of the downstream node on the path. If an s-d path is set up through SU-SU edges whose end nodes have fewer number of common available PU channels, it is more likely that the path will not be usable for a longer time (necessitating a path transition) due to chances of an upstream node running out of available PU channels that are also in the neighborhood of the downstream node on the path. An SU-SU edge whose constituent end nodes have a larger number of available PU channels could be used for a longer time (by making use of one of the common available channels for data transmission and reception). Thus, to maximize path lifetime, it is imperative that routing protocols give preference SU-SU edges that have a relatively larger number of common available PU channels to be part of an end-to-end SU-SU path.

This forms the motivation for our work in this paper. We propose a distributed routing protocol that selects the path with the largest value for the sum of the number of common available PU channels between two consecutive nodes (i.e., edges) of the s-d path in a network of SU nodes. Referred to as MCPUR (Maximum Common Primary User-channel Routing) protocol, the protocol aims at minimizing the number of path transitions and hence could be categorized as a stability-based routing protocol for CRAHNs.

We compare the performance of MCPUR with that of a shortest path routing (SPR) protocol that determines SU-SU paths based on only the objective of minimizing the number of hops and not giving preference to SU-SU edges whose end nodes have a larger number of common available PU channels. We observe MCPUR to incur a relatively fewer number of path transitions that could be as low as 62% (and 25% on average) to that of the path transitions incurred with SPR; the tradeoff being a slightly larger hop count per path (at most 17%; 8% on average). We thus show that in a complex dynamic network like that of CRAHNs, minimum hop based routing need not be the best routing strategy to determine stable paths that exist for a longer time. Indeed, the overhead associated with broadcast route discoveries (affecting the energy consumption and node/network lifetime) incurred due to frequent path transitions could be alleviated with the use of stability-based protocols like MCPUR. Hence, we claim MCPUR to be a significant contribution to the literature of distributed routing protocols for CRAHNs.

The rest of the paper is organized as follows: Section II presents the design of the proposed MCPUR protocol for CRAHNs and analyzes its effectiveness with an example. Section III presents a detailed simulation study of the MCPUR protocol vis-a-vis a SPR protocol for CRAHNs. Section IV discusses related work on routing solutions for CRAHNs. Section V concludes the paper and outlines directions for future research. Throughout the paper, we use the terms ‘route’ and ‘path’, ‘node’ and ‘vertex’ as well as ‘edge’ and ‘link’ interchangeably. They mean the same.

II. DESIGN OF THE MCPUR PROTOCOL

A. Network Model

Let $N_{PU}$ and $N_{SU}$ be the number of licensed primary users (PU nodes) and unlicensed secondary users (SU nodes) respectively. Let $R$ be the transmission range for the nodes and is the same for both the PU and SU nodes. Both the PU and SU nodes are assumed to be static. All the nodes are assumed to be uniform-randomly distributed in a 2-dimensional network of dimensions $[0..XMAX] 	imes [0..YMAX]$. We assume an SU node to know the location of the SU nodes and PU nodes in its neighborhood. An SU node and PU node are considered to be neighbors if the Euclidean distance between the two nodes is less than or equal to the transmission range, $R$. As part of the network start-up procedure, we let each SU node to broadcast its ID and the list of neighbor PU nodes to all the SU nodes within its transmission range, $R$. Such a broadcast facilitates an SU node to learn the identity of its neighbor SU nodes as well as determine the common PU channels with each of the SU nodes in its neighborhood (using the list of neighbor PU nodes broadcast by its neighbor SU node).

We model a time-invariant SU graph $G_{SU}$ of the CRAHN as the graph comprising of only the SU nodes; a link exists between two SU nodes $u$ and $v$ in $G_{SU}$ if and only if the Euclidean distance between the two SU nodes is less than or equal to the transmission range, $R$. We model a time-variant SU graph $G(SU)$ of the CRAHN as a snapshot graph of the CRAHN at time instant $t$; the graph comprises of both the SU nodes and PU nodes; there is no link between any two PU nodes; there exists a link between two SU nodes $u$ and $v$ in $G(SU)$ if and only if there exists a link between $u$ and $v$ in $G(SU)$ as well as there exists at least one common PU node whose licensed channel is available in the neighborhood of both $u$ and $v$ for use (i.e., the PU node is turned OFF). We take snapshots of the CRAHN for every $t_{sample}$ seconds and generate the snapshot graph $G(SU)$ of the network for every such sampling time instant.

For each of these time instants, we also identify the preferred PU channel that can be assigned for each SU node. The preferred PU channel for an SU node is the available PU channel whose corresponding PU node (is turned OFF) is in the neighborhood of a majority of the neighbors of the SU node. In case of a tie, we arbitrarily assign one among the contending PU channels as the preferred PU channel. We assign a preferred PU channel
per SU node (that is common to a majority of the neighbor SU nodes of the SU node) so that the number of channel switches per hop on an end-to-end SU-SU path could be minimized.

We assume there are can be a maximum of $N_{PU}$ distinct PU channels available for the SU nodes to be used as a data channel; a PU channel is simply identified by the ID of the corresponding PU node. An SU node is assumed able to be able to identify a PU channel based on its ID. A PU channel is turned ON and OFF alternatively for a random time period uniform-randomly chosen each time from the range $[0...\text{MAX}_{\text{Random\_ON}}]$ and $[0...\text{MAX}_{\text{Random\_OFF}}]$ respectively. In other words, a PU node decides to stay ON for a time period selected uniform-randomly from the range $[0...\text{MAX}_{\text{Random\_ON}}]$; after staying active for the duration of time selected, the PU node stays OFF for a different time period that is chosen uniform-randomly from the range $[0...\text{MAX}_{\text{Random\_OFF}}]$. The node again turns ON and stays active for a different time period chosen randomly from $[0...\text{MAX}_{\text{Random\_ON}}]$ and then goes back to OFF mode. This procedure is followed independently at each PU node throughout the duration of the network session.

We assume the presence of a common control channel [6] that is available for use by all the SU nodes. An SU node uses the common control channel to sense the local spectrum and identify the PU channels that are available for use. The SU nodes also use the common control channel for the propagation of the RREQ and RREP packets as well as the PU-UC (PU channel update) and RERR (Route error) packets as part of route discovery and maintenance respectively.

B. Route Discovery

Whenever a source SU node $(s)$ requires to start a data transfer session to a destination SU node $(d)$, the SU node initiates the route discovery process by broadcasting a Route Request (RREQ) packet to its neighbors through the common control channel that can be accessed by all the SU nodes. The broadcasts for the $s$-$d$ pair are identified using a unique ID. The RREQ packet includes three fields (see Figure 1) that are updated by the SU nodes on the path along which the packet propagates from the source to the destination: Route Vector field in which each intermediate forwarding SU node as well as the source/destination SU nodes record their ID; Preferred PU Channel Vector field that will be updated with the preferred PU channel at that time instant (one entry for each node) by the source and destination SU nodes as well as by each of the forwarding nodes of the RREQ packet; Number of Common Available PU Channels field that is updated by a downstream node of an SU-SU link indicating the number of common available PU channels with the upstream node of the link (updated by all the intermediate forwarding SU nodes and the destination SU node).

When an SU node $u$ receives the RREQ packet (for the particular broadcast session of the $s$-$d$ pair) from an SU node $v$, it checks if there is at least one common PU channel available in their respective neighborhood (i.e., there exists at least one PU node in the neighborhood of both $u$ and $v$ and that is currently turned OFF, available for use by the SU nodes) and also checks if its ID is already not listed in the Route Vector field of the packet. If only both the above validations return true, the SU node receiving the RREQ packet updates the Route Vector, the Preferred PU Channel Vector and the Number of Common Available PU Channels fields respectively with its SU Node ID, its preferred PU channel and the number of common PU channels available in the mutually intersecting neighborhood with the SU node from which the RREQ packet was received, and then rebroadcasts the RREQ packet in its neighborhood of SU nodes. An SU node simply discards an RREQ packet in which its node ID is already recorded (to avoid looping) and/or if there is no common available PU channel to the SU node from which the RREQ packet was received.

The destination SU node $d$ receives the RREQ packets along one or more paths from the source SU node $(s)$. For every RREQ packet received, the destination SU node updates the three fields in the packet with entries corresponding to itself. After waiting to receive the RREQ packets for a certain time (large enough to have received sufficient number of RREQ packets for the particular broadcast cycle), the destination SU node selects the best $s$-$d$ path as follows: The Route Vector field indicates the sequence of SU nodes on the path from the source to destination. Any two adjacent SU nodes in the Route Vector field constitute an edge of the path. The destination SU node simply adds the Number of Common Available PU Channels corresponding to the edges of the path traced by the RREQ packet. The destination chooses the route that yielded the largest value for the sum of the number of common available PU channels. In case of a tie, the minimum hop path among the contending paths is chosen; if the tie cannot be still broken, one among the contending paths is arbitrarily chosen.

The destination SU node $d$ initiates a Route Reply (RREP) packet back to the source SU node $s$. Since the links are bi-directional, the RREP packet could simply propagate on the reverse of the chosen $s$-$d$ path. The Route Vector and the preferred PU channel fields of the RREQ packet are copied in the RREP packet. An intermediate SU node receiving the RREP packet records in its routing table the source and destination end SU nodes of the path, the upstream and downstream nodes of the path (identified from the Route Vector field) as well as their preferred PU channels and forwards it further upstream towards the source SU node. The source SU node starts the data session after receiving the RREP.

C. Transfer of Data Packets and Channel Switch

The data packets for an $s$-$d$ session between two SU nodes $(s$ and $d)$ are routed along the $s$-$d$ path determined according to the route discovery procedure described in Section 2.2. An SU node remains tuned to its preferred PU channel for that time instant. To transmit a data
packet, the sending node (i.e., the upstream node of a hop) takes the responsibility to keep itself in sync with the preferred PU channel of the receiving node (i.e., the downstream node of the hop). If the preferred PU channels for both the upstream and downstream nodes of a hop are not the same, the upstream node switches its data channel to the preferred PU channel of the downstream node of the hop, and then switches back to its preferred PU channel (after the transmission across the link is completed). Thus, an SU node on the s-d path receives the data packet on its preferred PU channel; but may have to transmit the data packet on a data channel different from its preferred PU channel. This procedure is repeated at every link on the s-d path until the data packet reaches its destination d.

D. Route Maintenance

If the preferred PU channel for an SU node changes, it notifies all the upstream SU nodes of the s-d paths that it is part of through a PU Channel Update (PU-CU) packet sent on the common control channel. The upstream nodes accordingly update their routing table. Thus, any change in the preferred PU channel for a downstream node is locally handled without much overhead. When an upstream SU node finds out that it has no common available PU channel to its downstream SU node on the s-d path, it decides that the route has broken and notifies the source node s through a Route Error (RERR) packet. The source node s then initiates a fresh broadcast route-reply cycle to determine a new s-d path to the destination SU node d.

E. Example to Illustrate the Network Model, Route Selection and Effectiveness of MCPUR

Figure 2 presents a comprehensive example illustrating the network model and route selection for MCPUR vis-a-vis SPR as well as explains the effectiveness of MCPUR. The first column of the figure illustrates a time-invariant state of the network wherein all the SU nodes and PU nodes are present (the SU nodes are shown in larger circles with white color and the PU nodes are shown in smaller circles with a shaded color); the time-invariant graph G_i(SU) presents the superset of links that could be there between any two SU nodes. The second column of Figure 2 illustrates the state of the network at some time instant t. The PU nodes that are turned ON are shown in orange color; the PU nodes that are turned OFF are shown in light-blue color. The time-variant graph in the second column presents the network of SU nodes: there exists a link between two SU nodes if and only if a link exists between them in G_i(SU) and there is at least one common available PU channel in their shared neighborhood; the weight of the link is the number of available common PU channels; the numbers on the top of each SU node indicate the preferred PU channel for the node.

We explain the selection of the preferred PU channel for an SU node with an example: Consider SU node 7 that has links with two other SU nodes (SU nodes 2 and 3). SU node 7 has two available PU channels (PU nodes 1 and 2) in the neighborhood with SU node 2 and has one available PU channel (PU node 2) in the neighborhood with SU node 3. Thus, the SU node 7 chooses PU channel 2 that is available in majority among its neighbors. In Figure 2, we can also observe that following the above approach, PU channel 2 gets selected as the preferred PU channel for several other SU nodes too. As is evident in this example, the approach of selecting a preferred PU channel per SU node helps to minimize the number of channel switches; this along with the approach of selecting an SU-SU path with the largest sum for the number of common available PU channels helps to maximize path lifetime with the minimum number of channel switches.

The third column in Figure 2 illustrates the SU-SU path that gets selected between a source SU node (1) and a destination SU node (7) by following the procedure for MCPUR and SPR. Each intermediate SU node and the destination SU node attempt to be part of the SU-SU path through an edge that has a larger number of common available PU channels with its upstream node on the path. This approach of MCPUR path selection leads to path 1 - 6 - 4 - 2 - 7 whose sum of the # common available PU channels is 9 and the hop count is 4; on the
other hand, when we aim for a minimum hop shortest path, we find path 1 - 5 - 3 - 7 to be one of the minimum hop paths (hop count of 3), but its # common available PU channels is only 4. Such a path is likely to exist for relatively shorter time compared to the MCPUR path, as is also vindicated through our simulation results presented in Section 3.

III. SIMULATIONS

We conducted the simulations in the ns-2 (v. 2.35) discrete-event simulator [17]. As the focus of the paper is on the network layer (routing protocol design and performance analysis), we assume a perfect PU detection in the physical layer and an ideal MAC layer for transmission and reception of packets across each SU-SU link. The network is of dimensions [0...1000m][0...1000m]. The transmission range of any node (PU node and SU node) is 250m. We uniformly-randomly distribute the PU nodes and SU nodes in the network area. We conduct the simulations with 50 SU nodes and vary the number of PU nodes with values of 25, 40, 50, 75 and 100 (a fixed number of PU nodes for a particular simulation run). Accordingly, we introduce a parameter called the PU-SU ratio that is the ratio of the number of PU nodes to that of the number of SU nodes in the network; for the above said number of SU nodes and PU nodes, the PU-SU ratio ranges from 0.5 to 2.0. Each PU node is alternatively turned ON and OFF for a time period uniformly-randomly chosen from the range [0...MAX_{Random_{ON}}] and [0...MAX_{Random_{OFF}}] respectively. The values used for each of MAX_{Random_{ON}} and MAX_{Random_{OFF}} are: 5, 10 and 20 seconds. The network snapshot is taken for every 0.25 seconds; the total simulation time is 1000 seconds. A total of 30 s-d pairs were randomly chosen among the SU nodes. At any time, an SU node maintains a list of PU channels available in its neighborhood and is able to determine the presence/absence of one or more common PU channels with each of its neighboring SU nodes. As explained in Section 2.1, a comprehensive master list of all the PU neighbors per SU node is broadcast by the SU node at the time of network start-up. We assume the availability of a common control channel for the SU nodes to use for identifying the status of the PU channels as well as perform route discovery and maintenance.

We compare the performance of MCPUR with that of the minimum hop-based shortest path routing (SPR) protocol. SPR is designed and implemented similar to that of MCPUR (as described in Section 2); the only difference is that the destination SU node chooses the minimum hop path traversed by the RREQ packets and sends the RREP on the chosen minimum hop path (ties are broken arbitrarily). Both the SPR and MCPUR routes are considered to be valid as long as there is at least one common PU channel available for each pair of the end vertices of the constituent links of the path.

A. Performance Metrics

The following performance metrics were analyzed in the simulations:

(i) SU Network Connectivity: We measure the probability of connectivity of the network of SU nodes by running the Breadth First Search algorithm [7] on each of the time-variant graph snapshots G_t(SU) and evaluate the fraction (ranges from 0.0 to 1.0) of the total number of time instants the SU network is connected (i.e., there exists a path between any two SU nodes).

(ii) SU-SU Edge Ratio: This is the average of the fraction of the total number of SU-SU edges in the time-variant graph snapshots G_t(SU) to that of the time-invariant SU network graph G(SU), averaged across all of the connected SU network snapshots for the simulation runs corresponding to a particular combination of values for PU-SU ratio, MAX_{Random_{ON}} and MAX_{Random_{OFF}}.

(iii) Hop Count per Path: The time-averaged value of the number of hops per path, averaged over all the s-d paths and the duration of these paths.

(iv) # Path Transitions: We determine the average number of transitions per path (a path transition is defined as the change from one path to another path as a result of a broadcast route-request reply cycle), averaged over all the s-d sessions.

The simulation results presented in Figures 3 through 6 are averaged over 20 runs for each combination of values for PU-SU ratio, MAX_{Random_{ON}} and MAX_{Random_{OFF}}. Figure 4 illustrates the SU-SU edge ratio observed for scenarios with probability of SU network connectivity greater than 0. Figures 5-6 respectively report the values of hop count per path and # path transitions for probability of SU network connectivity values of 0.20 or above.

B. SU Network Connectivity

Figure 3 illustrates the values observed for the probability of SU network connectivity (ranges from 0 to 1). We observe that for a fixed value of MAX_{Random_{ON}} and MAX_{Random_{OFF}}, the probability of SU network connectivity increases with increase in the PU-SU ratio. Thus, for a given number of SU nodes and a fixed frequency of availability of the PU channels (fixed average duration of ON and OFF times for the PU nodes), the probability of connectivity of the SU network increases with increase in the number of PU nodes. For a fixed value of PU-SU ratio and MAX_{Random_{ON}}, the probability of connectivity of the SU network increases with increase in MAX_{Random_{OFF}}. This is as expected, as longer the duration of time the PU nodes are turned OFF, the longer the time the corresponding PU channels are available for use by the SU nodes. On the other hand, for a fixed value of PU-SU ratio and MAX_{Random_{OFF}}, the probability of SU network connectivity decreases with
increase in $MAX_{\text{Random,ON}}$ as the PU channels are likely to be unavailable for a longer time.

C. SU-SU Edge Ratio

The SU-SU edge ratio (for a particular operating condition of PU-SU ratio, $MAX_{\text{Random,ON}}$ and $MAX_{\text{Random,OFF}}$) is the ratio of the number of edges observed in the time-variant SU network graphs to that of the time-invariant SU network graph, averaged over all the time instants for which the time-variant SU network graph is connected across the simulation runs. For the SU node density considered in this paper (50 SU nodes with a transmission range of 250m distributed in a network of dimensions 1000m x 1000m: on average 10 SU neighbors per node in the time-invariant SU network graph), though an SU-SU edge ratio of 0.6 is sufficient to observe a non-zero probability of connectivity between any two SU nodes (i.e., probability of SU network connectivity $> 0$), in order to have at least a 20% chance of being connected, we require an SU-SU edge ratio of 0.88 or above.

![SU-SU Edge Ratio](image)

**Figure 4.** SU-SU Edge Ratio

D. Hop Count per Path

We observe the hop count per $s$-$d$ path to marginally increase (by about 8% on average; at most 17%) with maximum number of common PU channels-based routing compared to the minimum hop count per path encountered with shortest path routing. There is no particular operating condition in which we observe a larger difference in the hop counts; nevertheless, we observe the difference in hop count per path to be relatively more pronounced for $MAX_{\text{Random,ON}}$ values of 10 and 20. We attribute the marginal increase in the average hop count per path with MCPUR to the protocol aiming at finding paths that have a larger number of common available PU channels in the neighborhood of the constituent SU-SU links of the path and only using the hop count to break any ties. Since MCPUR incurs only at most 17% larger hop count per path and 8% on average (compared to minimum hop-based shortest path routing) and a significantly lower number of path transitions (as low as 62%, as observed in Section 3.5), we conjecture that the performance of MCPUR with respect to hop count will be comparable (if not better) to any other CRAHN routing protocol.

E. Path Transitions

We define a path transition as change from one path to another. We also consider the discovery of the first $s$-$d$ path as a path transition. We show the path transitions in a logarithmic scale (log to the base 2) in Figure 6. Only path transitions incurred for SU networks with a probability of connectivity 0.2 or above are reported. We observe MCPUR to consistently incur fewer path transitions than that of SPR for all the conditions. The percentage reduction in the number of path transitions is very much appreciable for almost all the scenarios, especially as the SU network connectivity increases. For a given value of $MAX_{\text{Random,ON}}$ and $MAX_{\text{Random,OFF}}$ the difference in the number of path transitions incurred by MCPUR and SPR shows a trend to increase with increase in the PU-SU ratio. Given that we increase the PU-SU ratio by keeping the number of SU nodes a constant and by only increasing the number of PU nodes, we could infer that as the number of available PU channels increases, the two protocols start showing diverse performance. MCPUR makes use of the increase in the number of PU nodes and their unique channels and discovers end-to-end $s$-$d$ paths between two SU nodes that are likely to exist for a longer time. We can also observe that the absolute number of path transitions incurred for both MCPUR and SPR decreases with increase in the PU-SU ratio (for a fixed $MAX_{\text{Random,ON}}$ and $MAX_{\text{Random,OFF}}$) - indicating that the availability of a larger number of PU nodes and their channels in the neighborhood of the SU nodes lets the end-to-end SU-SU paths to stay for a longer time, even in the case of shortest path routing.

For a given PU-SU ratio and $MAX_{\text{Random,ON}}$, the absolute number of path transitions incurred by both MCPUR and SPR decreases with increase in $MAX_{\text{Random,OFF}}$. This is due to the virtue of an increase in the availability time of the PU channels for SU use. When complemented with a larger number of PU channels (an increase in the PU-SU ratio), a larger value of $MAX_{\text{Random,OFF}}$ facilitates better SU network connectivity and reduced number of path transitions for both MCPUR and SPR. MCPUR makes use of the increase in the availability time of the PU channels more effectively by discovering $s$-$d$ paths that are relatively more stable than those discovered using SPR.

For a given PU-SU ratio and $MAX_{\text{Random,ON}}$, the absolute number of path transitions for both SPR and MCPUR increases with increase in the value of $MAX_{\text{Random,ON}}$. This is because, as the PU nodes tend to use their channels for a longer time, the SU nodes could not use those PU channels and have to frequently switch from one PU channel to another in search of an available PU channel. For a given PU-SU ratio and $MAX_{\text{Random,OFF}}$, the difference in the absolute number of path transitions for both MCPUR and SPR shows a trend to decrease with increase in $MAX_{\text{Random,ON}}$. This is because as the PU channels stay active for a longer time, the number of common available PU channels in the neighborhood of the SU nodes decreases; the route selection criteria of MCPUR cannot be that effectively enforced when the number of common available PU channels for use is very limited - as a result, the difference in the number of path transitions for MCPUR and SPR reduces. Nevertheless, we observe MCPUR to be able to incur a significantly lower number of path transitions than that of SPR for larger $MAX_{\text{Random,ON}}$ values.
Routing solutions for cognitive radio ad hoc networks can be broadly categorized into two classes: full-spectrum knowledge based and local spectrum knowledge based. The full-spectrum knowledge based routing solutions (e.g., [8] [9]) are typically centralized in nature and assume that each SU node knows the global knowledge about the availability of the PU channels in the entire network; these solutions are used to derive at benchmarks for performance with respect to one or more routing metrics. The local spectrum knowledge based routing solutions (like MCPUR) are distributed in nature and rely only on the spectrum information gathered in the neighborhood (through the common control channel). The local spectrum knowledge based routing solutions proposed so far could be classified into sub classes that target at optimizing a particular metric in a distributed fashion.

The minimum power routing protocol [10] is designed to discover SU-SU paths that incur a lower energy consumption by taking into consideration the energy loss incurred due to transmission, reception, broadcast route discovery as well as channel switching. We opine that MCPUR will be complementary to the minimum power routing protocol when used together as a hybrid protocol, we expect optimal performance with respect to all of the above metrics (aided by fewer path transitions). The bandwidth footprint (BFP) minimization-based routing protocol [11] attempts at discovering an s-d path that will minimally impact the on-going s-d sessions with respect to the interference area of the SU nodes (called the bandwidth footprint). In [12], the authors evaluated the tradeoffs associated with farthest neighbor routing (FNR) and nearest neighbor routing (NNR) for CRAHNs; results indicate FNR to achieve better end-to-end channel utilization and reliability and NNR to be relatively more energy-efficient.

In [13-16], the authors attempted to develop delay-sensitive routing protocols for CRAHNs. While [13, 14] focus on minimizing the sum of the channel switching and access delays at the intermediate nodes, [15] focuses on minimizing the sum of the queuing delays at the intermediate nodes. In [16], the authors propose a load-balancing strategy wherein an intermediate node that is located on more than one s-d path (between a given source s and destination d) locally coordinates with its neighbors and (based on its current load) decides whether or not to redirect the received s-d data packets through these neighbor nodes.

As can be seen from the above discussion of related work, there is no single work that has proposed the use of the number of common available PU channels (in the mutually intersecting neighborhood of the SU nodes constituting the links of a source-destination path) as the criteria to select stable routes that are likely to go through fewer path transitions. Though there have been attempts at individually minimizing either the end-to-end delay per packet or energy consumption, MCPUR is the first such attempt to discover stable routes that can contribute towards accomplishing optimal values for both end-to-end delay per packet and energy consumption as well as provide better quality of service due to less frequent changes in the paths traversed by the data packets.
minimum hop count incurred with SPR and the average difference (increase) in hop count is only about 8% (and the difference could be at most 17%). On the other hand, MCPUR consistently incurs fewer path transitions than SPR for all the simulation conditions and the difference could be as large as 62%. We are thus confident that the performance of MCPUR will be comparable to that of any other existing CRAHN routing protocol and at the same time incur a relatively lower number of path transitions. Future work has been planned with regards to extending MCPUR for spectrum underlay-based CRAHNs (the network graph of which is likely to have unidirectional links) and mobile CRAHNs (where the PU nodes and SU nodes are mobile).

ACKNOWLEDGMENTS

This work is funded through the Massie Chair Grant (#: DE-NA0000654) at Jackson State University.

REFERENCES


