Computer Communications 32 (2009) 1983–1997

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

Search: A routing protocol for mobile cognitive radio ad-hoc networks

K.R. Chowdhury [a,*], M.D. Felice [b,1]

[a] School of Electrical and Computer Engineering, Georgia Institute of Technology, 75 5th Street, Suite 5158, Atlanta, GA 30332, USA
[b] Department of Computer Science, University of Bologna, Bologna, Italy

ARTICLE INFO

Article history:
Available online 28 June 2009

Keywords:
Ad-hoc networks
Cognitive radio
Mobility
Routing
Spectrum allocation

ABSTRACT

Recent research in the emerging field of cognitive radio (CR) has mainly focussed on spectrum sensing and sharing, that allow an opportunistic use of the vacant portions of the licensed frequency bands by the CR users. Efficiently leveraging this node level channel information in order to provide timely end-to-end delivery over the network is a key concern for CR based routing protocols. In addition, the primary users (PUs) of the licensed band affect the channels to varying extents, depending on the proportion of the transmission power that gets leaked into the adjacent channels. This also affects the geographical region, in which, the channel is rendered unusable for the CR users. In this paper, a geographic forwarding based SpEctrum Aware Routing protocol for Cognitive ad-Hoc networks (SEARCH), is proposed that (i) jointly undertakes path and channel selection to avoid regions of PU activity during route formation, (ii) adapts to the newly discovered and lost spectrum opportunity during route operation, and (iii) considers various cases of node mobility in a distributed environment by predictive Kalman filtering. Specifically, the optimal paths found by geographic forwarding on each channel are combined at the destination with an aim to minimize the hop count. By binding the route to regions found free of PU activity, rather than particular CR users, the effect of the PU activity is mitigated. To the best of our knowledge, SEARCH takes the first steps towards a completely decentralized, CR routing protocol for mobile ad-hoc networks and our approach is thoroughly evaluated through analytical formulations and simulation study.

1. Introduction

The emerging field of cognitive radio (CR) networks is geared to address the increasing congestion in the unlicensed band by opportunistically using vacant spectrum, such as, frequencies licensed for television broadcast, public service, among others [1]. While there has been considerable research effort in devising efficient spectrum sensing and sharing algorithms at the node level, it is important to seamlessly integrate these designs in the implementations of the end-to-end network protocols. As an example in a CR network, routes constructed at the network layer must not affect the ongoing transmission of the primary users (PUs) of the licensed spectrum and thus, they must have an awareness of the spectrum availability. Moreover, when a PU is detected, the routing protocol must make the key decision of either (i) switching the channel in the affected portion of the route, or (ii) passing through entirely different regions altogether, thus increasing the latency. The frequently changing PU activity and the mobility of the CR users make the problem of maintaining optimal routes in ad-hoc CR networks challenging. In this paper, we propose the SpEctrum Aware Routing for Cognitive ad-Hoc networks (SEARCH) protocol based on geographic routing, that adapts to the dynamic spectrum availability and the node mobility, while trying to maintain end-to-end connectivity.

In this paper, we make the following contributions for routing in CR networks.

1.1. Awareness of primary user activity

Existing routing protocols for ad-hoc networks do not account for the region affected by an active PU and are unaware of the changing spectrum opportunity. In CR networks, routes must be constructed to avoid these regions and must also adapt themselves to subsequent changes induced by new PU arrivals. We discuss this further for geographic routing protocols, as the route formation in SEARCH is based on this principle.

Geographic routing protocols for classical wireless networks rely on a greedy positive advance towards the destination [4]. By knowing the location of the destination and that of the candidate

* Corresponding author. Tel.: +1 404 510 9089.
E-mail addresses: kaushikc@ece.gatech.edu (K.R. Chowdhury), difelice@cs.unibo.it (M.D. Felice).
[1] This work was completed when the author was a visiting researcher at the Broadband Wireless Laboratory, Georgia Institute of Technology, Atlanta, GA 30332, USA.

0140-3664/$ - see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2009.06.011

forwarding nodes within its range, a node can choose the next hop that gives the greatest advance. Whenever a void is encountered, the path traverses the perimeter of the void and then resumes the greedy forwarding to the destination. However, in a CR network, as the PUs have precedence in accessing the spectrum resource, the routing protocol must compromise on the greedy advance when it intersects a region of PU activity. Two possible solutions to this are: circumventing the affected region or switching the affected channel. The choice between these options needs to made taking into account the overall end-to-end performance. Classical geographic routing protocols are oblivious to these PU specific considerations and limit their operation to a single channel. SEARCH, on the other hand, provides a hybrid solution. It first uses greedy geographic routing on each channel to reach the destination by identifying and circumventing the PU activity regions. The path information from different channels is then combined at the destination in a series of optimization steps to decide on the optimal end-to-end route in a computationally efficient way. Instead of binding the path to a set of forwarding nodes, it is described by *anchor* locations, i.e., regions found free of PU activity. Moreover, the route management function in SEARCH modifies an existing route with minimal overhead whenever a new PU is detected.

1.2. Global channel switching vs path detour decisions

From practical considerations, the channels are not completely orthogonal and a finite amount of transmitted power leaks into the adjacent channels. This constitutes the spectral leakage power, which is a source of the interference in the affected channels. Based on the proportion of the leakage power, channels used by the CR may be affected to different geographical extents due to a single PU. As an example, consider Fig. 1, in which, a WLAN transmitter is located at the origin O and follows the IEEE 802.11b standard [13]. The $x-y$ axes represent the two dimensional cartesian space and the z axis is the frequency scale. S_1 and S_2 are two other nodes that sense the transmitted power and are at a distance of ℓ_1 and ℓ_2, respectively, from the WLAN where $\ell_1 > \ell_2$. The nodes S_1, S_2 and the WLAN lie in the $x-y$ plane. When the WLAN transmits on channel 6, the nodes observe the leakage power by tuning to the channels $Ch : 4,\ 5,\ 7$ and 8, each separated by a difference of 5 MHz. The spectral shape of the WLAN signal is such that the leakage power is high in the frequencies close to that of the transmission channel, and falls as one moves further away on the frequency scale. The received power also falls exponentially with increasing distance on the $x-y$ plane. Consequently, the *coverage radius* of the transmitter, i.e., the distance up to which the channel power serves as a source of interference, depends upon the (i) frequency difference in the channels used for transmission and power measurement, and the (ii) distance between the transmitter and the sensing node. This radius is shown by $R_4, \ldots, R_8$ for the WLAN channels $Ch : 4, \ldots, 8$, respectively. We observe that node S_1 is further away from the origin, as compared to S_2, and is within the coverage radius of the WLAN on channel 6 only. Node S_2, being closer to the WLAN, is under the coverage range of all the channels considered above.

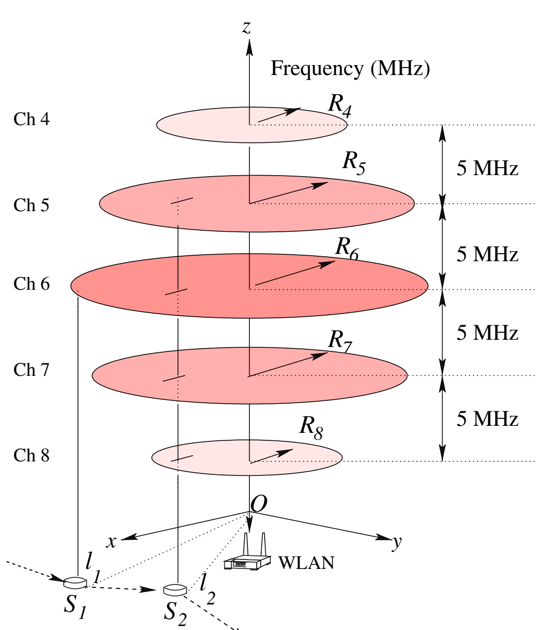

Fig. 1. Different coverage regions in different channels.

Extending this example for CR networks, a single PU located at the origin may affect the channels used by the CR users, say S_1 and S_2, differently. As the extent of the coverage is not completely known at a given node, a local channel switching decision may not be optimal in a network-wide context. From Fig. 1, consider a path that passes through the nodes S_1 and S_2. Node S_1 finds itself under the PU coverage region on channel 6 and switches to a free channel, say 5. The subsequent route, however, intersects the PU coverage region in the new channel within a few hops, when node S_2 is reached. In addition, the node S_1 does not have any information about the extent of the detour that will be required if the route formation is continued on the affected channel 6. SEARCH solves this problem by estimating the route detours on each channel, taking into account their dissimilar coverage ranges. It then computes whether the cost of the detour is less than the added delay in switching the channel, with respect to the end-to-end latency.

1.3. Cognitive radio specific mobility concerns

The node mobility is one of the chief factors of route outages in ad-hoc networks. Specifically, in a CR environment, we identify a new problem arising out of mobility that needs to be addressed. Consider the case in which the nodes continue to form a connected route but may stray, due to their mobility, in the coverage region of a PU on the current channel. Thus, even though the route is connected, the nodes in the PU region may be unable to transmit. We recall that SEARCH uses *anchor* locations for routing, that are found to be free of PU activity. It addresses the CR mobility concerns by checking if the next hop node is within a threshold distance of the anchors. If this condition is not satisfied, the forwarding node identifies a new next hop that is closest to the anchor, thus ensuring that PU regions are always avoided.

Geographic routing protocols rely on the knowledge of the location of the destination for route computation. When the destination itself moves with time, the current route needs to be extended by progressively adding nodes at the end. SEARCH predicts the destination location in advance through Kalman filtering approaches, so that the route is suitably extended and packets are reliably delivered to the new location.

The rest of this paper is organized as follows. Section 2 describes the related work in this area, while the assumptions of our approach are given in Section 3. In Section 4, we present the route setup phase of SEARCH, our CR routing protocol in detail. Section 5 describes the route maintenance functions related to the changing spectrum and node mobility. A thorough performance evaluation is conducted in Section 6. Finally, Section 7 concludes our work.

2. Related work

Routing is a well researched area in classical ad-hoc networks with protocols designed for diverse mobility considerations, optimization constraints and hardware assumptions. SEARCH is designed for CR networks and differs from the general class of protocols for ad-hoc networks in its consideration of (i) regions affected by PU activity, (ii) route optimization undertaken over several channels, and (iii) awareness of the dynamically changing channel environment. In this section, we specifically focus on existing centralized and distributed routing protocols for CR networks. We also discuss approaches based on the principle of geographic routing, which is an integral component in SEARCH.

The centralized framework presented in [12] defines a *link disruption probability* to simulate PU activity. It is assumed that this probability is known before network operation through some estimation techniques. The routing problem is formulated as an optimization goal that minimizes the total average delay, which is determined by: (i) the volume of data that must be sent, (ii) the mean capacity of the path as a function of the spectrum and link disruption probabilities, and (iii) channel propagation delay. The centralized nature of the algorithm that needs complete network topology makes this approach infeasible in mobile ad-hoc networks. Besides, the link disruption probability for a given node pair cannot be assumed independently of the others as all nodes within a PU coverage region experience similar disruption in their communication. Another centralized approach [16] addresses flow fairness in which, the complete knowledge of the flows between any two nodes is known. Here, a network-wide optimization problem is solved and a constant factor approximation to the optimal solution is provided. A graph theoretic approach is given in [7], wherein non-overlapping channels and time slots over a two hop range are assigned through the creation of independent sets if the entire network graph is known. Two routing algorithms, that require the network topology at each node, are provided in [2] for optimizing the path latency and number of channel switches, respectively. In [6], a *layered graph* is constructed which has the available channels in vertical layers, and the classical network graph showing the node adjacencies along each layer. The nodes within communication range on a given channel (layer) have a horizontal edge between them. All the channels that are usable at a given node are connected by a vertical edge. The costs of an edge traversal is appropriately set depending on the time cost of switching a channel (vertical edge) as against the forwarding of the packet on the same channel (horizontal edge). The problem of routing can now be expressed as finding the shortest path in this modified graph any classical techniques may be used for this. However, in mobile networks, collecting the entire network topology is infeasible. This approach serves as the optimal centralized solution for routing algorithms with static topologies.

Distributed approaches scale well in an ad-hoc network and require comparatively less computation at the destination. However, a key problem faced in the design of distributed routing protocols is that the path and channel decisions are made sequentially and not together. The best routing paths are first identified and *then* the preferred channels along the path are chosen in [3]. Here, the ad-hoc on-demand distance vector (AODV) routing protocol is modified to include the list of preferred channels by a given node as the route request (RREQ) is forwarded through the channel. Once the RREQ is received, the destination is aware of the channels that may be used to transmit at each hop and finds the optimal combination such that channel switching is minimized. However, during the route setup, this RREQ is transmitted over the control channel and *not* on the channels actually used for routing. Thus, the arrival time of the RREQs is not reflective of the actual performance of the route in the channels eventually used for routing. Moreover, the condition that there may not be any available channels in certain segments of the route is not considered.

The route formation in the SEARCH protocol is based, in part, on geographic routing. This principle is used in GPSR that undertakes greedy forwarding under normal conditions and enters into perimeter mode when a void (region with an absence of forwarding nodes) is encountered [4]. In order to circumvent this void, it requires the construction of a planar graph of the network at each node at all times or the creation of network-wide spanning trees [5]. However, this constitutes an overhead as only a few selected nodes need to participate in the perimeter forwarding mode. the classical GPSR has also been modified for particular application scenarios, such as, mobile vehicular networks in GPCR [8] and GPSRJ++ [9]. Though these efforts feature improved route maintenance ability under mobility assumptions and overcome the need to maintain the planar graphs, they need street level knowledge of roads and junctions, thus restricting their application. Mobility considerations are addressed in [10,11] based on local single-hop information. Through periodic beacon exchanges, a node signals if it is in a *dead-end* region and prevents itself from being chosen as the next hop during greedy forwarding in [10]. By estimating the velocity based on location updates through beacon messages, a node can estimate when the next hop will go out of range in [11]. Similarly, the authors claim that significant packet loss occurs at the penultimate node in the path if the destination has moved from the original location, but is still present within its transmission range. By simply checking for the destination node at each hop, before greedily forwarding the packet, some resilience to destination related mobility is achieved in [11]. However, this neither provides a solution to a large scale destination mobility, nor reevaluates the optimality conditions of the current path periodically, unlike our proposed SEARCH protocol. In summary, protocols based on geographic forwarding are currently devised for single channel networks and have no support for CR specific issues.

SEARCH is a completely distributed routing solution that accounts for PU activity, mobility of the CR users and jointly explores the path and channel choices towards minimizing the path latency. It reflects on the true channel delay by conducting the route exploration in the actual channels used for data transfer. We now list the assumptions regarding the network architecture in the next section.

3. Network architecture

In our CR mobile ad-hoc network nodes may move freely in the two dimensional cartesian space. The PUs are assumed to be stationary and they coexist in an overlapping region with the CR users. Each PU operates with an ON–OFF switching cycle that is unknown to the CR network. In this section, we describe the network architecture by considering the node and channel aspects of the network in detail.

3.1. Node architecture

We assume that a CR user can tune its radio transceiver to any of the allowed channels in the licensed (primary) band. Each node has a single radio and there is an underlying channel coordination mechanism at the link layer that allows neighboring nodes to engage in pairwise communication. Several works have addressed this issue [17,18] and in this paper, we limit the scope to the routing protocol for CR networks. We assume that the source node is initially aware of the location of the destination, though the latter may periodically send position updates once the route is in operation. Each node is also aware of its own location through GPS or triangulation techniques [19]. There are no prior assumptions of the number, locations, transmission standard or protocol that is followed by the PU.

3.2. Channel structure

The primary band comprises of n possible CR channels, each having a bandwidth $B_i,\ i = 1, \ldots, n$, that are known in advance. The PU activity is modeled as a two-stage ON–OFF process. We consider the general case, in which, the PU activity results in a spectral overlap with the CR channels, which in turn, reduces significantly as one moves further away from the PU's transmission frequency. We consider the overlap factors as 1, 0.5 and 0.25 for the CR user channels that are at 0, 1 and 2 channel spacings away from the PU's central channel frequency. These values can be easily modified for specific network environments. The spectral leakage power gives the different channel coverage regions, described in Section 1, that must be avoided. The CR users undertake periodic sensing and can reliably detect the presence of a PU. Both the location and the sensing information is broadcast by the CR user, through beacons, to the other nodes within its transmission range.

We now describe our spectrum aware routing protocol, SEARCH, for CR networks.

4. Search: a CR routing protocol

SEARCH attempts find the length of the shortest path based on greedy advancement that may be traversed on a combination of channels to the destination. The key functionality in our proposed approach is evaluating when the coverage region of the PU should be circumvented, and when changing the channel is a preferred option. First, the shortest paths to the destination, based on geographic forwarding and consideration of the PU activity, are identified on each channel. The destination then combines these paths by choosing the channel switching locations, with an aim to minimize the number of hops to the destination. In this section, we shall describe the initial protocols functions in two parts: (i) the route setup phase and the (ii) route enhancement that is undertaken to improve the route, once it is in operation. The route maintenance functions that deal with route outages due to PU arrival and node mobility are covered in detail in Section 5.

4.1. Route setup

In this stage, a route request (RREQ) is transmitted by the source on each channel that is not affected by the PU activity at its current location. It gets forwarded by intermediate hops till it reaches the destination, with each intermediate node adding in the packet its (i) ID, (ii) current location, (iii) time stamp and (iv) a flag status indicating the current propagation mode of the algorithm. SEARCH operates in two modes – *Greedy Forwarding* and *PU Avoidance*, depending on whether the RREQ is propagating along the greedy shortest path to the destination or needs to circumvent a region of PU activity, respectively. Finally, the routes on the individual channels are combined at the destination by the *joint channel–path optimization* algorithm.

4.1.1. Greedy forwarding

Through the exchange of beacon messages, each CR user is aware of the locations of the other nodes within its transmission range. Thus, knowing the location of the destination, a protocol based on greedy geographic forwarding can decide which of the candidate forwarders of the RREQ should be chosen as the next hop to minimize the distance to the destination. While SEARCH shares the principle of geographic forwarding with other related ad-hoc protocols like [4], the next hop is *not always* chosen purely on the greedy advance metric in our approach. The two distinguishing features are:

- The RREQ forwarding process must occur on the same channel and the chosen next hop must not be under a PU coverage region on the current transmission channel. As an example, from Fig. 1, node S_2 cannot participate in the route formation on the channels $4, \ldots, 8$. It may, however, forward the arriving RREQs on the other channels that are not affected by the PU transmission.
- The chosen forwarder must lie in a specific region around the current hop, called as the focus region, which we define later in this section. Thus, a node with a lesser advance towards the destination but within the focus region is chosen over another node closer to the destination that lies outside this area.

In Fig. 2, the source S has the nodes x, y and z within its transmission radius R_T. These nodes are at a straight line distance of l_x, l_y and l_z, respectively, from the destination D, where $l_x > l_y > l_z$. The focus region for S is shown by the sector $S - AB$ and extends to an angle of θ_{max} from the line SD. Classical geographic routing protocols like GPSR would have chosen node z at this stage, while SEARCH chooses the node with the greatest advance *within* its focus region, i.e., node y. If no such node exists, then SEARCH switches from the greedy forwarding phase to the PU avoidance phase.

We now define the focus region formally with reference to Fig. 2.

Definition 1. *Focus region:* Consider a straight line path, SD from a given node S to the destination D. The sector with transmission radius R_T, centered at the location of the node S and extending up to a maximum angular spread of θ_{max} on either side of SD, gives the focus region for this node.

Definition 2. *Decision point:* A node that lies in the focus region of the previous hop along the path, but does not find a forwarding node for the RREQ in its own focus region, is called the decision point.

The focus region is shown as the shaded area in Fig. 2. If the nodes x and y do not participate in the routing process, then node S would not have any candidate forwarder within its focus region. It would mark itself as a decision point, as per Definition 2.

Knowing the decision point (DP) gives an intuitive idea of the locations of the PUs and the occupied channels. We recall that the nodes within the coverage region of a PU do not take part in the RREQ forwarding process. Thus, they are virtually absent from the network topology on the affected channels and do not feature in the focus

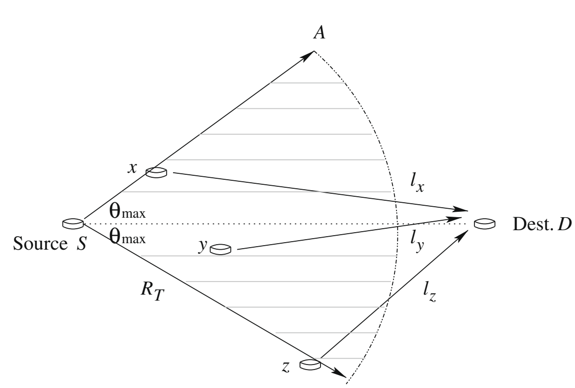

Fig. 2. Using greedy geographic forwarding on a given channel.

region as candidate forwarder nodes. Not finding any feasible node, this previous hop now labels itself as the DP and enters into the PU avoidance stage. When the RREQ is received at the destination, it knows the point on the route, i.e., the DP location, from which the path enters into a detour to avoid the PU region. SEARCH then attempts to find out alternate paths (and hence channels) at these detour locations.

While our approach may decrease the per-hop advancement, it allows SEARCH to identify a large deviation from the straight line path to the destination. We note that if a void (region where no nodes are present) is encountered, the same process is followed. Thus, unlike GPSR [4], SEARCH does not need to maintain the planar graph information when it encounters regions that need to be avoided.

4.1.2. PU avoidance

When a PU region is encountered, rendering the channel in its vicinity unusable, the greedy forwarding towards the destination can no longer be carried out. This stage is called the PU avoidance stage as the RREQ now starts circumventing around the affected region. We explain this as follows:

Fig. 3(a) shows the shaded circular area under the influence of a PU on the channel being used for forwarding the RREQ. In addition, the focus region for node x on this channel, from Definition 1, is given by the sector $x - AB$ with the maximum angle of $2 \cdot \theta_{max}$. Some of the nodes that sense the PUs and do not participate in the forwarding of the RREQ, lie in the focus region of the node x. Through the periodic beacon update, these affected nodes inform their one-hop neighbors, including node x, of the current state of the channel environment. Thus, node x is aware that the closest node to the destination that can forward the RREQ (node a) lies outside its focus region. From Definition 2, node x concludes that it is a DP and sets the PU avoidance (PA) flag in the RREQ packet before retransmitting it. The DP marks the point from which the route must circumvent the region of PU activity on the given channel. There may be several such DPs in the path to the destination and this information is collected by the RREQ as it traverses through the network. The PA flag in the RREQ remains set till a node is reached that has a candidate forwarder in its focus region. In the example shown in Fig. 3(b), the RREQ traverses the node a, b and finally reaches node c. The latter has a candidate forwarder, node d, that lies in its focus region. At this point, i.e., at node d, the PA flag is reset, signaling the end of the avoidance phase and the greedy forwarding is resumed.

4.1.3. Joint channel–path optimization

If the licensed band is completely free of PU activity, then there are no regions that must be avoided in any of the channels. Thus, the RREQs sent on the different channels would chart similar paths to the destination. In the presence of PUs, however, some nodes do not participate in the route setup stage if they are in the affected region on these channels. Consequently, the paths traversed by the RREQs in the different channels are not the same. The optimization phase of the SEARCH protocol is designed to choose a combination of channels and the propagation paths along them that minimize the end-to-end latency. This optimization is performed at the destination based on the information contained in the received RREQs. We first explain the underlying idea by the following example and then formally define the optimization algorithm.

In Fig. 4, each plane represents a channel and the broken line shows the path obtained by the propagation of the RREQ on each of them. This path is limited on one channel and is a combination of the greedy forwarding and PU avoidance phases. Though all the nodes are located in the same physical space, only a subset of them are shown that participate in the routing on a given channel, depending on their individual PU coverage regions. These nodes form the *anchor* points on their respective channels.

Let the RREQ on channel 1 give the shortest path from the source to the destination D, among all the considered channels. A portion of this path from node x to D is shown. Let node x be the decision point (DP) where the path starts avoiding the PU affected region on channel 1. Our optimization framework tries to identify if a better channel (and hence, path) may be used at the DP, thereby preventing the additional hops incurred in channel 1. To allow a path switch at node x, the new path must have the node x in common, or must be within the transmission range of x. Thus, only the paths in channels 2 and 3 are considered as the nodes y and z are within range of x on these channels, respectively. The path on channel 4, being at a distance of at least two hops from node x cannot be included in this stage. The feature in SEARCH that allows distant paths to be combined is described in Section 4.2.

If the path on channel 1 is switched to a different channel, say channel 2, then the packet traverses on the path in channel 1 till node x and then on the path on channel 2 from node y to the destination. The channel, and hence, path change occurs at node x if the sum of the (i) path latency in the new channel from node y to the destination, (ii) the estimated time to reach node y from node x, and (iii) the cost of the switching the channel t_s is less than

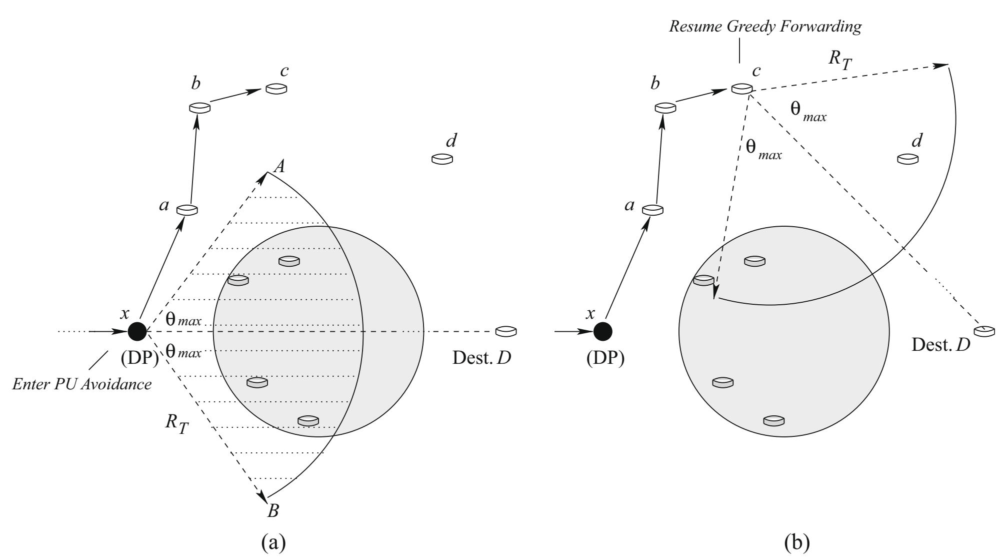

Fig. 3. The PU avoidance phase with the focus region.

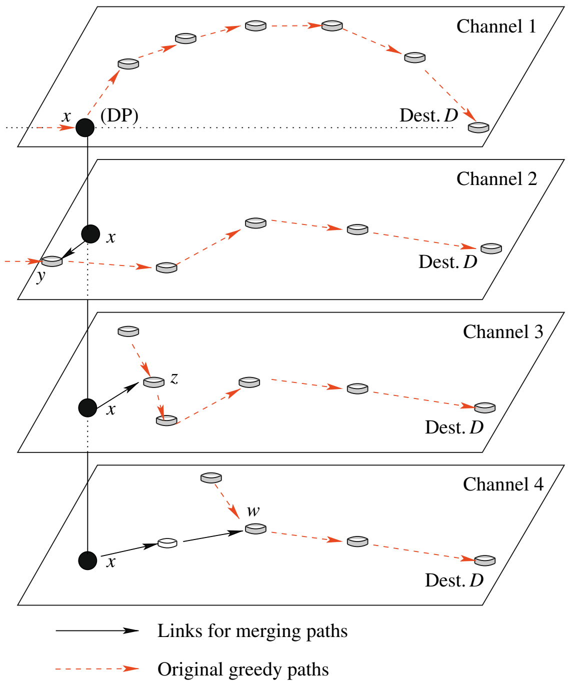

Fig. 4. The joint path and channel decisions at destination.

the path delay from node x to the destination in channel 1. As the new path is traversed, the above conditions are checked for each new DP encountered in the route. If multiple channels exist that provide a better path alternative, one of them is chosen that minimizes the total latency to the destination. From Fig. 4, the time to destination may be smaller for channel 2 as compared to continuing on channel 1 or switching to channel 3, and thus, it is chosen at the node x. A brief analysis of these two options of route detour as against channel switching is presented for an ideal case in Appendix A.

We define the variables used in the subsequent discussion as follows: The RREQ received on a given channel, say k, defines a path of l_k anchor points given by $P_k = \{A_k^1, A_k^2, \ldots, A_k^{l_k}\}$. The anchor points are the locations of the nodes that were chosen during the forwarding process of the RREQ packet and indicate the absence of PU activity. Some of these anchors may also serve as decision points (DP) depending on whether the path encounters a PU region. The optimal greedy path P_G is a combination (or union) of anchor points and the channel switching decisions shown by $C_j^{k,i}$, indicating that the channel is changed from k to i at the hop j. The transmission time for a packet on channel i is given as T_R^i and the total number of channels in the band is C.

We can formally express the route selection at the destination through the following algorithm steps.

- *Step 1. Initial path selection:*The destination D receives the RREQs on each channel $k \in C$ and extracts the path information P_k from it. The path P_k comprises of a set of nodes $\left\{A_k^j\right\}$ with their respective time stamps $\left\{t_k^j\right\}$, where $j = 1, \ldots, l_k$ and the final node is the destination, i.e., $A_k^{l_k} = D$. We first define the latency, L_k^m, from an intermediate point m to the final node l_k on the channel k based on their respective time stamps, t_k^m and $t_k^{l_k}$, as follows:

$$L_k^m = t_k^{l_k} - t_k^m \tag{1}$$

As the first optimization step, considering the propagation from the first node of the route ($m = 1$), the least latency path L_k^m is chosen among all the available channels $k \in C$ using Eq. (1),

$$i = \arg_k \min\left\{L_k^1,\ \forall k \in C\right\} \tag{2}$$

In this step, the greedy path solution P_G is initialized to the start node on the channel i as $P_G = A_i^1$. This set will grow as the choice of nodes and channel switching decisions are added progressively in the subsequent steps of the algorithm.

- *Step 2. Greedy path formation:*The least latency path may be further improved by switching to a different channel at the DPs. This is because the initial path having the minimum end-to-end delay, may not be continuously optimal through the intermediate path segments. In this step, SEARCH attempts to improve the chosen route by considering intersecting paths on the other channels that may be locally optimal. The DPs are chosen for this optimization as the path starts curving at these locations and better paths on different channels that follow a shorter route to the destination may exist. Formally the next hop node, $A_i^{j=j+1}$, along the current chosen path on channel i is added to P_G, if it is not a DP. This step is repeated in a loop as long as a DP is not reached. Thus,

$$j = j + 1$$
$$P_G = P_G \bigcup A_i^j, \quad \text{if } A_i^j \neq \text{DP} \tag{3}$$

If this next hop is the final destination, then the algorithm is terminated and the path is complete. In this case, SEARCH proceeds to Step 5 directly. If this is not so, and the next hop is indeed a DP, the local optimality condition is checked in Step 3.

- *Step 3. Optimization at the DP:*When a DP is reached, SEARCH attempts to find an *intersecting* path on a different channel at this location. Each node, during forwarding of the RREQ, includes its own location information in the packet. Thus, the destination knows of the locations of each of the nodes that the RREQ traverses through. Thus, it simply uses these location coordinates and the known transmission range to infer which of the nodes, if any, in the other paths are within the reach of a given node. A given path P_1 with node x is said to be *intersecting* with another path P_2, if the latter (i) has the node x common or (ii) has a node that is within transmission range of node x in path P_1. From Fig. 4, the paths shown in channels 2 and 3 are intersecting with the path on channel 1 as the nodes y and z in these paths, respectively, are within transmission range of the node x. Given the channel switching time t_s and the time to transmit a packet T_R^k on channel k, the time overhead, δ_k, to reach the node in the intersecting path is,

$$\delta_k = t_s + T_R^k, \quad k \in C \tag{4}$$

The current channel i (and hence, the path) may be switched to an intersecting path (say, on channel k) at the node m in the transmission range of the DP j, only if the latter has a smaller latency to the destination measured from j. Also, the transmission range (R_T) constrains the allowed distance between the two nodes A_i^j and A_k^m by the inequality $dist\left(A_i^j, A_k^m\right) < R_T$. The total time taken to reach the destination is given by $L_k^m + \delta_k$, if channel k is chosen. Also, if more than one intersecting path exists, the new channel k is so chosen that the time to destination is minimized over all the possible channel options. Assuming the transmission range as R_T, the tuple {hop number m', channel k'} is chosen that minimizes the total time $L_k^m + \delta_k$,

$$\{m', k'\} = \arg_{\{m,k\}} \min\{L_k^m + \delta_k\} \text{dist}\left(A_i^j, A_k^m\right) < R_T, \quad \forall k \in C \tag{5}$$

- *Step 4. Route expansion:* The greedy path solution is updated with the new channel and path information. First, the channel switching decision shown by $C_j^{i,k'}$, is incorporated in the final path P_G, along with the node $A_{k'}^{m'}$ that serves as the next hop in the new path.

$$P_G = P_G \bigcup C_j^{i,k'} \bigcup A_{k'}^{m'} \tag{6}$$

  Finally, the new channel k' is now the default channel, i.e., $i = k', j = m$. The procedure of traversing the new path by checking the DPs is repeated from Step 2.
- *Step 5. Route confirmation:*When the last hop i.e., the destination is reached, the route reply RREP is sent back to the source along the optimal route P_G. The RREP contains the IDs of the nodes, the anchor locations and the channel switching decisions. The routing of data packets can begin immediately when the source receives the RREP.

During the route setup stage, the destination optimizes the path and channel choices based on nodes that are within transmission range of each other to give the optimal path P_G. The next stage of optimization, route enhancement, occurs when the route is active and attempts to combine paths that are separated by more then one hop.

4.2. Route enhancement

The route enhancement stage comes into operation *after* the initial route setup stage and conservatively explores the gains of linking together paths formed on different channels that are up to η hops away. From Fig. 4, if the DP x is a part of the optimal route at the end of the route setup phase, this stage of the protocol may allow it to reach node w two hops away, on the least latency path in channel 4. The algorithm for enhancing the current route is as follows:

- We recall that the route setup phase checks for a possible path (and channel) change at each DP. If there is no other node within the transmission range, the DP is retained in the current path. In this phase, the currently used shortest path P_G is further optimized considering (i) all the remaining DPs on it and (ii) the anchor points on the other routes (hence, on the other channels) that are within η hops of the considered DP.
  We assume that the currently used optimal path comprises of a set of anchor points given by $P_G = \{A_G^1, \ldots, A_G^q\}$. Formally, SEARCH chooses the DP on the optimal path (A_G^i) and the anchor A_k^m on channel k, that must be reached, to minimize the total distance to the destination. Here, the first constraint is that the node A_k^m must be within η hops of the DP A_G^i, i.e., $\text{dist}(A_G^i, A_k^m) < \eta \cdot R_T$. The maximum allowed physical distance between the two nodes is given by the product of the hop count η and the transmission range, R_T. On similar lines, the actual number of hops, $\eta_{i,m,k}$, between these two nodes is given by,

$$\eta_{i,m,k} = \frac{\text{dist}\left(A_G^i, A_k^m\right)}{R_T} < \eta \quad \text{where} \quad A_G^i = \text{DP} \tag{7}$$

  If T_R^k is the transmission time for a packet on channel k' for a single hop, the total estimated time taken to traverse this distance of $\eta_{i,m,k}$ hops is $\eta_{i,m,k} \cdot T_R^k$. In order to find the tuple $\{i', m', k'\}$ that minimizes the total cost to the destination among all the possible combinations, we add the time to the destination from the new next hop mL_k^m and the channel switching time t_s. Additionally, this time has to be smaller than the path latency, L_G^i, from the considered DP to the destination if the routing is continued on the existing path.We formulate this optimization equation to find the tuple $\{i', m', k'\}$ as follows:

$$\begin{aligned} &\{i', m', k'\} = \arg_{i,m,k} \min \left\{ \left(\eta_{i,m,k} \cdot T_R^k + L_k^m + t_s \right) < L_G^i \right\} \\ &\forall A_G^i \in P_G, \text{ and } A_G^i = \text{DP} \\ &\forall k \in C, \\ &\forall A_k^m \in P_k, \\ &\eta_{i,m,k} < \eta \end{aligned} \tag{8}$$

  Summarizing the above discussion and from Eq. (7), we explain the constraints as follows: A node A_G^i in the optimal path P_G may be used in the minimization only if it is a DP. We consider the path P_k in each of the possible channels k in the channel set C. In these paths we allow the anchor points as a candidate solution if it is within η hops of the DP being considered.
- Once a feasible path is identified by the destination, it sends a route enhancement (ROP) message to the DP $A_G^{i'}$, found in the earlier step. The ROP contains (i) the ID of the node $A_{k'}^{m'}$ that must be reached on the new path, (ii) the path information in terms of successive next hop nodes from the intended node $A_{k'}^{m'}$ to the destination. We recall that both of this information is obtained from the RREQ received on channel k' during the route setup stage.
- The DP $A_G^{i'}$ receiving the ROP now sends an RREQ message with the node $A_{k'}^{m'}$ as the destination, similar to the route setup phase involving greedy forwarding. The new path information received in the ROP is then included in the RREP when it is forwarded on the channel k'. A key difference in the route enhancement phase is that, on receiving the RREQ, node $A_{k'}^{m'}$ continues to forward it towards the eventual destination D. The RREQ may be dropped if either the target node $A_{k'}^{m'}$ cannot be reached in η hops, or the route (indicated by the node IDs in the RREQ) is disconnected at any point before the destination is reached.
- If the destination receives the forwarded RREQ, it checks if the actual latency on the new path is lower than the value on the current path from the DP $A_G^{i'}$. If so, an RREP is sent along the new route and an RERR is propagated along the earlier route indicating the formation and the teardown of the routes, respectively. In this process, the portion of the optimal path P_G after the DP $A_G^{i'}$ is deleted and the new path information is added. Thus,

$$\begin{aligned} &P_G = P_G \Big/ \left\{ A_G^{i'+1}, \ldots, A_G^D \right\} \\ &P_G = P_G \bigcup \left\{ A_{k'}^{m'+1}, \ldots, A_{k'}^D \right\} \end{aligned} \tag{9}$$

- This run-time optimization continues till all the DPs in the current optimal path P_G, have been explored and no further improvement in latency is observed.

We note that this optimization process does not affect the current route operation as it is carried out in parallel. However, this stage is implemented *immediately* after the route is operational as mobility of the nodes may change the path conditions found in the route setup phase for the other channels. We next describe the route management functions of the SEARCH protocol that attempts to keep the route connected in the presence of the dynamic channel and mobility conditions in CR networks.

5. Route maintenance

This phase of the SEARCH protocol addresses the following concerns:

- *PU awareness:* The appearance of a PU may render the region in its vicinity unsuitable for routing. In such cases, the nodes in the affected regions must immediately cease operation in the occupied channel and, if needed, explore alternate routes to the destination.

- *CR user mobility:* Node mobility may cause route disconnections. Even if the route stays connected, nodes may stray into PU activity regions and cause undesirable interference to the licensed users. Moreover, the source and destination may themselves move and the earlier route, formed on the basis of their relative geographical locations, can no longer be considered optimal.

The route management phase allows SEARCH to adapt to the above changes in the channel and network environment with minimum delay as follows:

5.1. PU awareness

During the ongoing routing process, a node always checks if the next hop node is within its focus region. This condition becomes false if a new PU is detected by the next hop and it informs its neighbors of this event through beacon messages. At this time, similar to the PU avoidance phase described in Section 4.1.2, the node marks itself as a DP. It stops further routing of data packets and sends out an RREQ to the destination. The RREQ undergoes the same forwarding process in the route setup but with the following key differences: (i) It is sent by the newly formed DP instead of the source node and (ii) it carries as a payload, the node and channel decisions of the earlier path from this DP to the destination. The RREQ is propagated on the current channel till either the destination is reached or one of the nodes common to the previous operational path is encountered. In Fig. 5, the existing route is shown by the bold line Source(S) $-$ U $-$ DP_1 $-$ A $-$ B $-$ DP_2 $-$ V $-$ Destination(D). The two decision points DP_1 and DP_2 are the nodes where the route curves around an existing PU affected region. The points U and V represent nodes within one-hop transmission range of the source and destination, respectively. The vertical displacements indicate channel switching along the frequency axis, while the $x-y$ axes represents the physical space. When the shaded region, intersected by the current route, becomes affected due to PU activity, node A identifies itself as a new DP. It sends out the RREQ and encloses in it, the IDs of the nodes B, DP_2, V, D. The detour paths taken by the RREQ are shown by the broken lines. If the RREQ follows the path P_1, it finds the node B in common with the earlier path and continues along the known hops to the destination. However, if it follows path P_2, it reaches the destination along the same channel, possibly following a more circuitous route. In either case, the destination sends back an RREP to the DP A if the latency is within a threshold L_{Th} of the original path or else, an RERR is propagated back to the source S signaling the need of a fresh route formation. Similarly, an RERR is sent back by the DP A to S if it does not receive a response to the transmitted RREQ within a pre-defined timeout period.

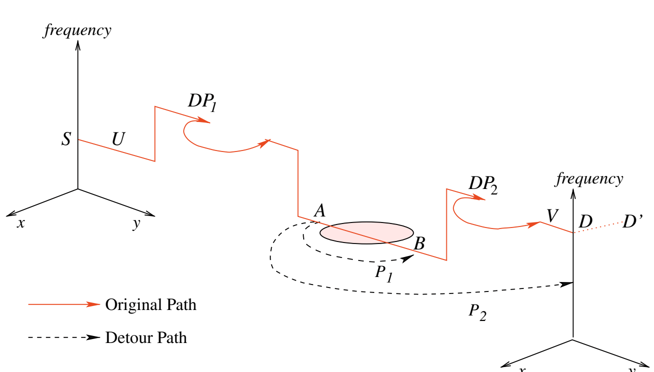

Fig. 5. Route maintenance with PU awareness.

5.2. CR user mobility

SEARCH does not associate the route with particular nodes in the network, but rather with *anchor* points that represent regions free of PU activity. We recall that each node, through periodic beacons, updates its one-hop neighbors about it current location. As long as the new node location is within a threshold distance of the anchor point found during route formation, it is retained as the next hop. If indeed this threshold is exceeded, then the ID of the next hop is removed the forwarding table and the entry is replaced by another node that is the closest to the corresponding anchor point. The binding of the route to PU activity free regions ensures continued route operation in place of node mobility. SEARCH also addresses separately the case of source and destination mobility, as they are specific end-points and cannot be replaced by other nodes close to the anchors. It extends the route at the source or the destination end, whenever they become unreachable. In our approach, the destination saves the information about its previous locations (in the form of $x-y$ coordinates) at regular time intervals. It then uses Kalman filtering to predict the new location and updates the previous hop node of its anticipated future position [20]. In this method, the states (given by the locations) are used to make the prediction of the next state, as well as the error covariance. As the destination moves to the new area, based on the difference between the true and predicted locations, the Kalman gain is updated, which in turn gives a new covariance matrix for the next prediction.

In Fig. 5, the destination D predicts its new location D' and informs the previous hop V. In the event that D becomes unreachable, node V sends out an RREQ with this predicted location on the same channel used between the two nodes. Once the RREP is received from the destination, the data forwarding is resumed. Similarly, the source S sends out an RREQ directed to the first node U in the current path, if it moves out of range. If the RREP is received, the route is extended and the routing of the data packets can be resumed. In either case, if no RREP is received, a fresh route setup is invoked.

The above route management techniques may add new nodes to the existing path through local decisions. This may result in the current routing path growing sub-optimally over time, especially in highly mobile scenarios. In SEARCH, the number of hops originally used for route formation and the hops used in the current path are periodically checked at the destination. If the difference is greater than the allowed threshold hop count, it signals the need of a new route formation by propagating the RERR message back to the source.

We next present the performance evaluation of SEARCH considering its different features and CR specific scenarios.

6. Performance evaluation

In this section, we evaluate the performance of the SEARCH protocol under different network conditions, traffic loads and mobility factors with the parameters as listed in Table 1. The simulation model is built in the NS-2 simulator with multi-radio multi-channel extensions [15]. We model the primary users' activities by using the exponential ON–OFF process described in Section 3. Simulations are performed in random multi-hop network topologies, in which, 400 nodes, unless specified otherwise, are distributed in an area of $1000 \times 1000\ \text{m}^2$. The coverage range of the PU on its occupied channel is 300 m and the transmission range of the CR user is set at 120 m. The node and PU locations are randomly chosen in each trial run and an average of 50 trial runs is used for a data point. In the mobility tests, the random waypoint model is used with a mobility of 2 m/s and the channel switching time is 5 ms [21].

Table 1
SEARCH simulation parameters.

Number of CR users	400
Number of PUs	[2,...,10]
Max node speed	2 m/s
Active connections	1
Traffic data type	UDP-CBR
SU Tx range	120 m
PU Tx range	300 m
Channel switching time	5 ms
Latency threshold (L_{Th})	0.2 s
Initial source–destination separation	850 m
Route optimization parameter η	2

We demonstrate the performance improvement attained by SEARCH, by (i) direct comparison with GPSR, suitably extended for a multi-channel environment and (ii) by selectively enabling the joint path and channel optimization modules present in it. We describe the terminology used for the simulation study as below:

1. *GPSR:* The classical GPSR [4] protocol is extended for a multi-channel environment. GPSR is oblivious to the PU activity and is run over all the available channels separately. The least latency path (from the source to the destination on the same channel) is considered as the final route chosen at the destination. We also incorporate the modifications proposed in [10,11] and described in Section 2, for alleviating the disruptions caused by node mobility.
2. *SEARCH (Least Latency):* In this version of the SEARCH protocol, the route setup phase with the greedy forwarding and the PU avoidance components are retained, as described in Sections 4.1.1 and 4.1.2, respectively. However, the path-channel optimization (Section 4.1.3) as well the route enhancement (Section 4.2) carried out at the destination is disabled. The final route chosen is the one that provides the least latency over all the available channels without any intermediate switching of channels. This serves as the benchmark for comparing the fully functional SEARCH, which switches channels at the appropriate locations in the route.
3. *SEARCH (Optimized):* Here, all optimization modules are enabled and we assume $\eta = 2$ for the route enhancement function (Section 4.2). In addition, both the optimized and least latency versions of SEARCH have the route maintenance functionality active, as described in Section 5.

We perform four different tests in our study by measuring the impact of (i) number of channels, (ii) number of CR users and the wait time at the destination for the RREQs, (iii) varying

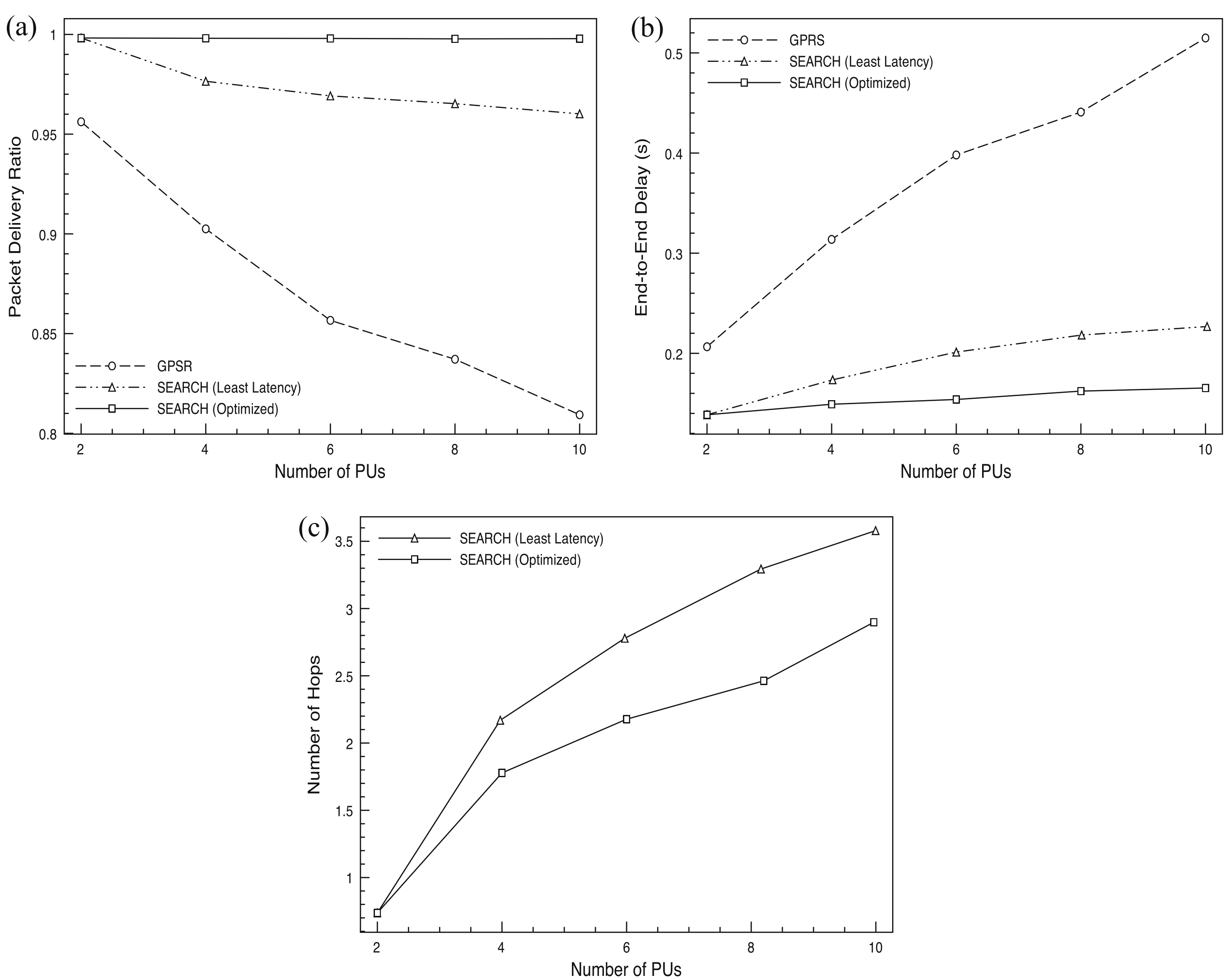

Fig. 6. The packet delivery ratio, the end-to-end latency, and the number of hops for the case of 5 channels are shown in (a), (b) and (c), respectively.

traffic load, and (iv) route maintenance under node mobility and PU activity.

6.1. Effect of number of channels

We consider two separate cases of 5 and 10 channels, in which, a randomly chosen number of PUs is considered from the range [1,10]. Whenever a PU transmits, two adjacent channels on either side of the central transmission channel are affected based on the spectral leakage ratio given in Section 3.2. The PU is kept in the ON state for the duration of this experiment and the source and destination are initially separated by a distance of 850 m.

We first consider the packet delivery ratio (PDR) for 5 and 10 channels, as shown in Figs. 6 and Fig. 7(a), respectively. We see a marked difference in PDR for the GPSR and the SEARCH protocols as the former does not account for the PU activity regions and may pass through them for the greatest advance to the destination. Apart from the effect on the CR user, a low PDR also implies that the PU reception is affected due to concurrent transmissions. Interestingly, SEARCH (least latency) does show a drop in the PDR as compared to the optimized case when PUs are increased. This effect is seen to a greater extent when 5 channels are present as compared to 10 channels. This is because with the increasing number of PUs renders large regions ineffective for transmission, even on the best available channel. Thus, there are some portions of the route that must intersect these PU affected region where detours are no longer possible. When more channels are present, the PU activity is shared among them and the region influenced by the PUs on any given channel is reduced. The ability to switch the channel in the optimized SEARCH allows the flexibility to maintain a high PDR, even under increasing number of PUs.

The difference in the end-to-end delay between GPSR and the SEARCH protocols is significant, as seen in Fig. 6 and Fig. 7(b). This is counter-intuitive as the path to the destination for GPSR is the least in terms of hops, as it is not sensitive to the presence of the PUs. Further investigation revealed that the total link layer delay caused by packet re-transmissions at each hop is significantly larger than the latency due to path detour in SEARCH, thus resulting in significant time saving for the latter. Moreover, the difference between the two flavors of SEARCH demonstrates the benefit of optimizing the path over several channels. Results reveal that the optimized channel switching gives nearly 60% improvement in the optimized SEARCH over the single channel least latency configuration.

We define the *path optimality* metric as the difference in the number of hops between the optimal shortest path if the global topology is known, as against the path that is found by the SEARCH protocol. In either case, the PU affected region is considered. In

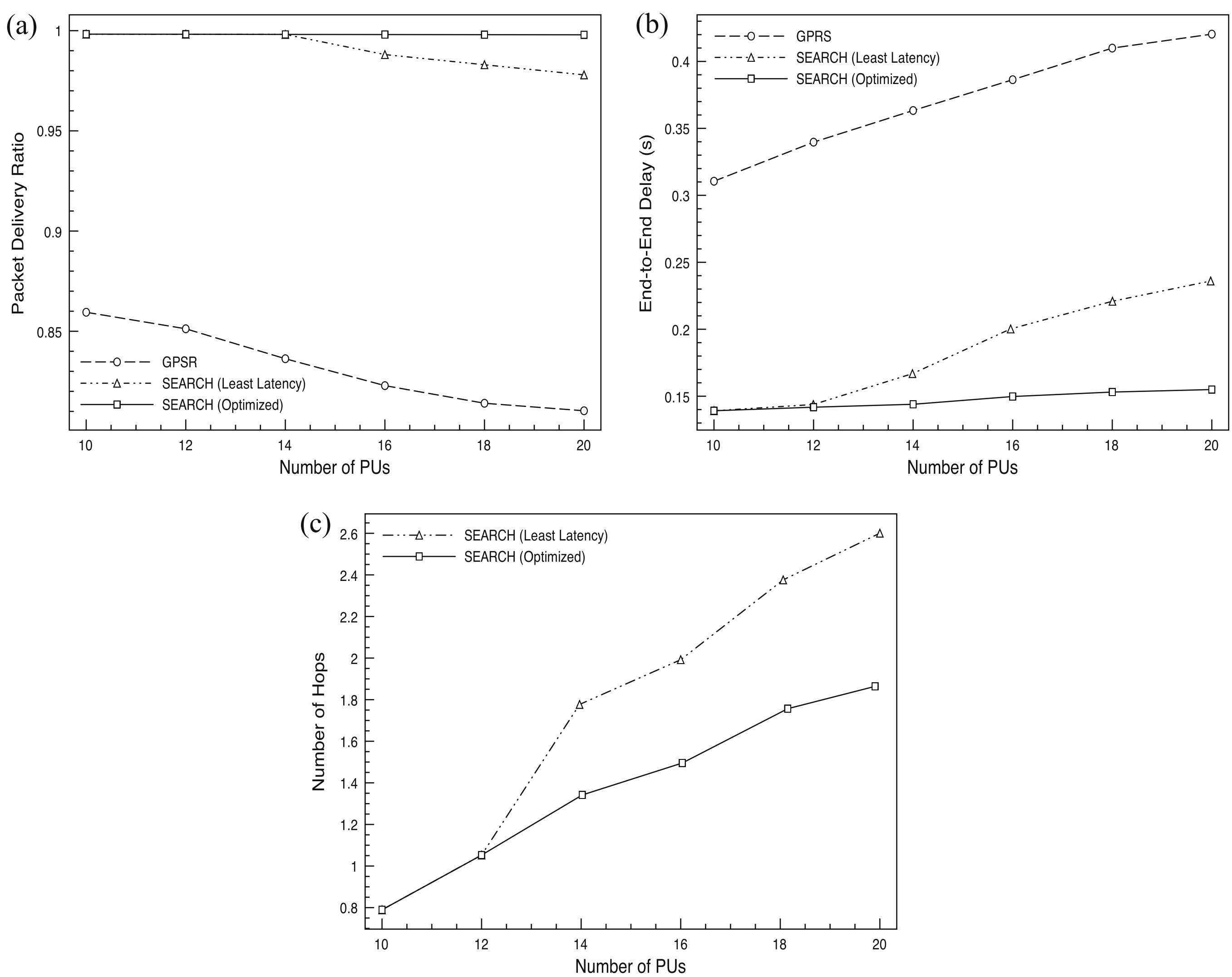

Fig. 7. The packet delivery ratio, the end-to-end latency, and the number of hops for the case of 10 channels are shown in (a), (b) and (c), respectively.

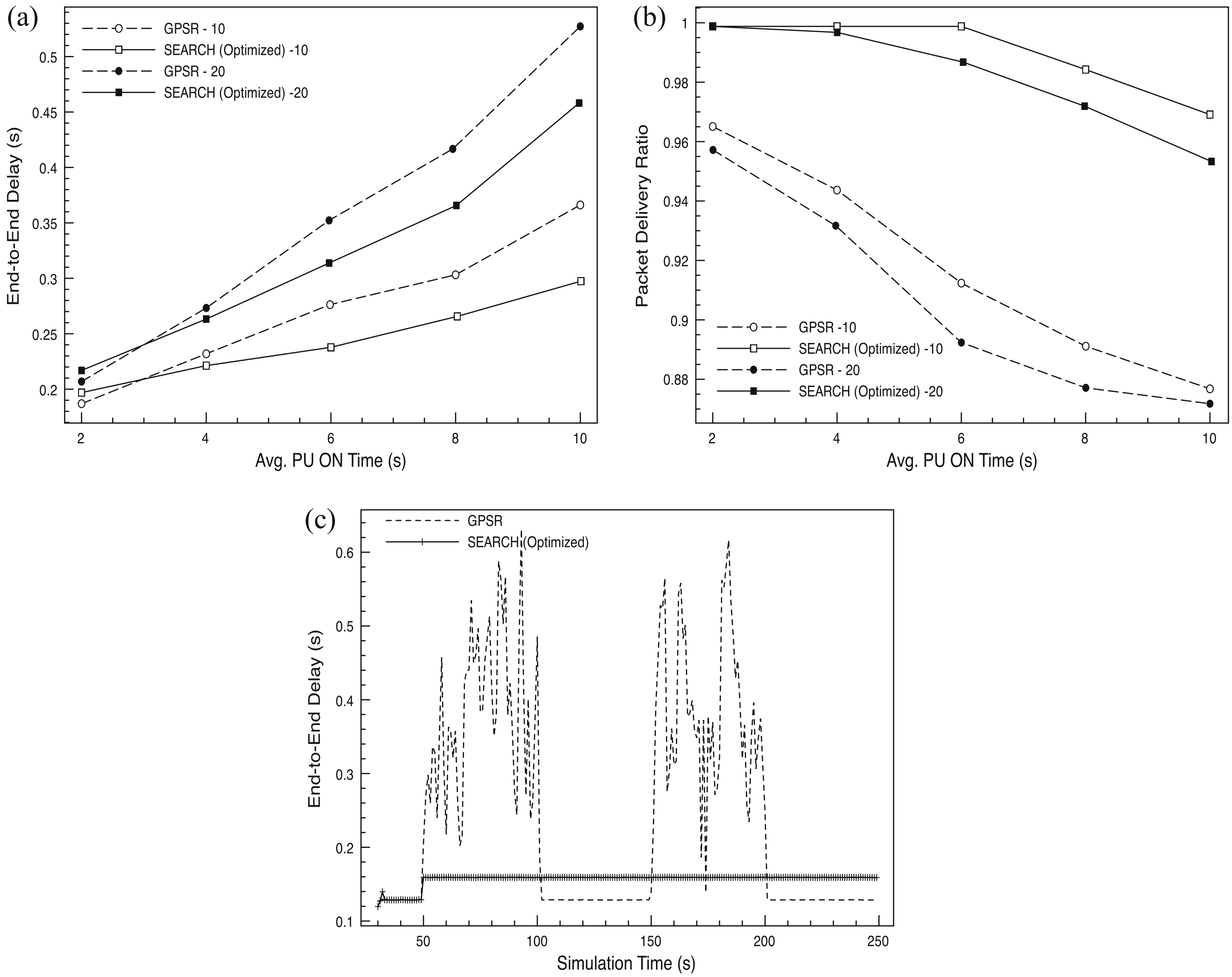

Fig. 8. The end-to-end latency and the packet delivery ratio are shown in (a) and (b), respectively, for different PU ON times. A snapshot of the latency is plotted against the simulation time in (c).

Fig. 6 and Fig. 7(c), we show this metric for 5 and 10 channels, respectively. Here we measure the difference in the number of hops of the two SEARCH protocols with the route that is constructed with the global topology knowledge. We observe that the SEARCH gives a good performance, within 4 hops of the optimum path. In addition, we observe that the number of channels affects the path optimality condition to a greater extent than increasing the PUs in the network.

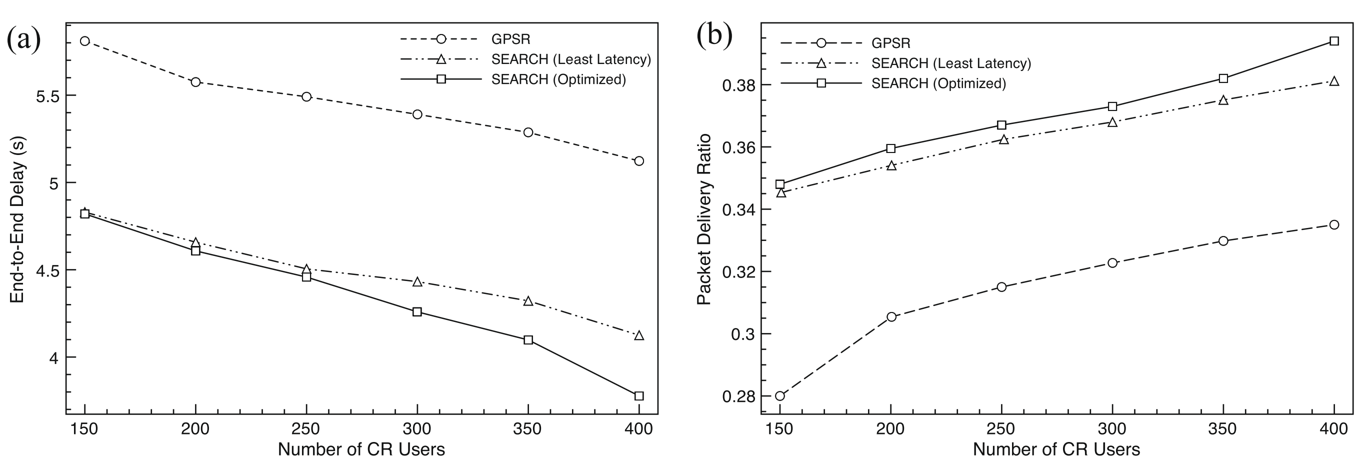

Fig. 9. Analysis of the impact of varying PU–CR user ratio is shown in terms of latency (a) and PDR (b).

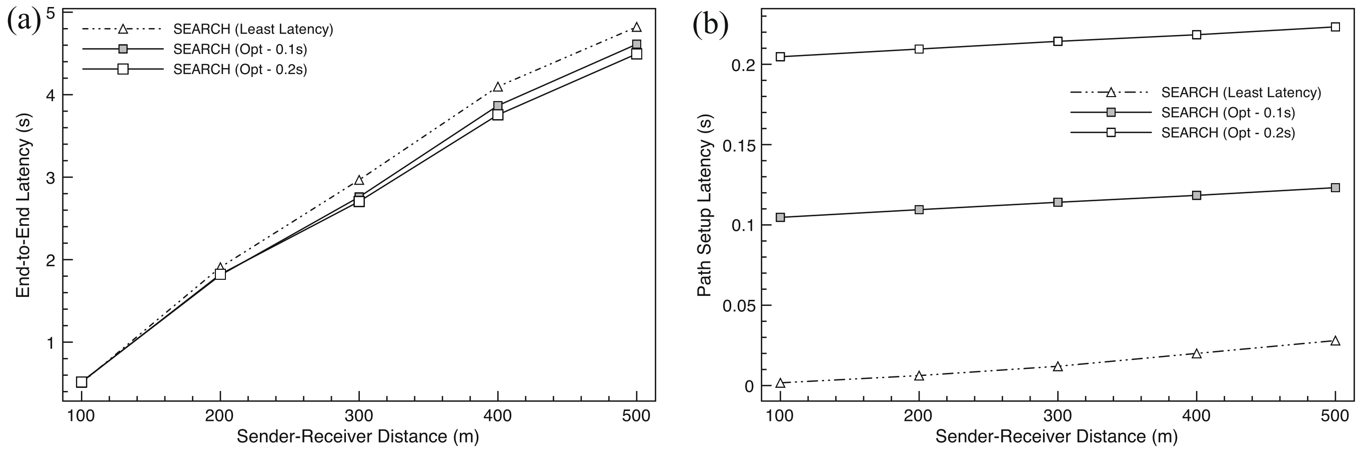

Fig. 10. Analysis of the wait time on end-to-end latency (a) and route setup latency (b).

6.2. Effect of CR users and RREQ wait time

We next observe how the ratio of the PUs to the CR users in the network affects the system performance by varying the latter from 150 to 400 keeping the number of PUs constant at 10. When we increase the number of CR users, we also increase the possibility of discovering new routes not affected by PUs, for all the evaluated protocols. As a result, the end-to-end delay (Fig. 9(a)) decreases for all the protocols with a significantly improved performance by the flavors of the SEARCH protocol. The PDR has a gradual fall with increasing PU–CR ratio, as seen in Fig. 9(b), thereby proving the scalability of SEARCH in a highly dense PU environment.

In Fig. 10(a), we observe that the end-to-end latency is considerably shorter for the optimized SEARCH protocol when the destination waits for 0.2 s to receive the RREQs, as compared to the other case for 0.1 s. Interestingly, though the route setup time is much larger (Fig. 10(b)) for the larger wait time, it is well compensated by identifying a better route.

6.3. Effect of traffic load

In the load analysis, we consider 10 PUs inside the network, but vary the system load by modifying the traffic produced by the source node measured in Kbps. Three connections for the CR users are active at any given time. We observe that the end-to-end delay Fig. 11(a) is considerably lower for the case of SEARCH optimized. The least latency SEARCH protocol, as well as GPSR, suffer from self-contention among packets of the same flow as only one channel is used for data forwarding and no channel switching occurs. This is the key reason for the rapid decline in the PDR for higher connection load for these protocols Fig. 11(b). In comparison, the optimized SEARCH protocol uses path segments on different channels whenever PU regions are encountered. This alleviates the problem of self-contention at the link layer thus giving a better PDR.

6.4. Route maintenance

In this study, we evaluate the ability of the SEARCH scheme to address the network changes arising from (i) the variable PU activity and (ii) node mobility, based on the route management functions described in Section 5.

6.4.1. PU activity

Fig. 8(a) and (b) show the end-to-end delay and the PDR, in a scenario with 5 and 10 PUs following the exponential ON–OFF process (Section 3). There are a total of 5 channels. The duty cycle for the switching model is varied in the range [0.5,2.5] s. When the PU activity is low, the GPSR [4] protocol produces the best performance in terms of end-to-end delay. This is because the shortest path is chosen by the greedy approach and there are no detours to avoid regions of PU activity. However, the performance of the GPSR [4] scheme degrades when the average ON time increases due to the interference caused by the PUs. Fig. 8(b) confirms that

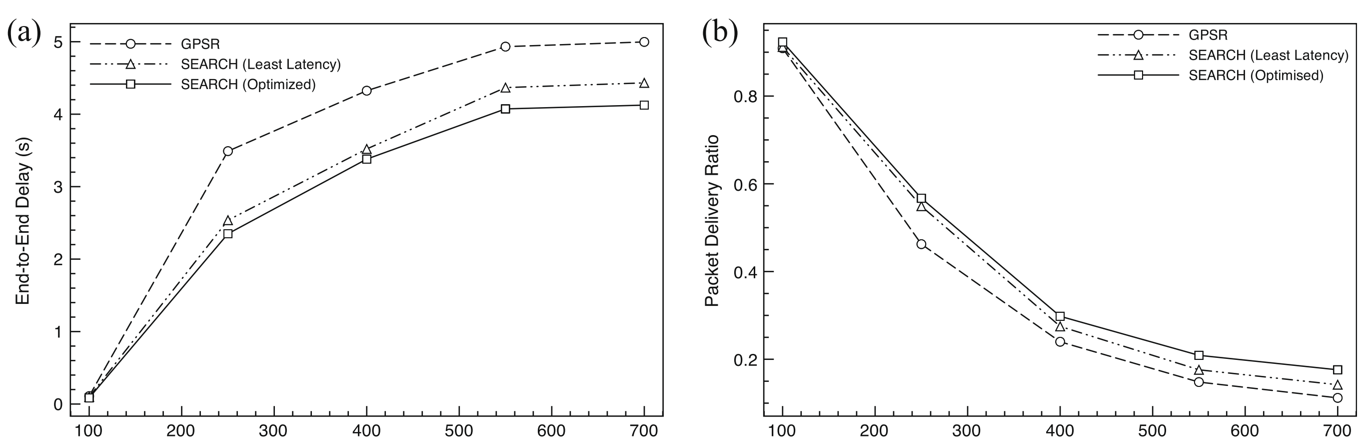

Fig. 11. The effect of connection load on the end-to-end latency and the packet delivery ratio are shown in (a) and (b), respectively.

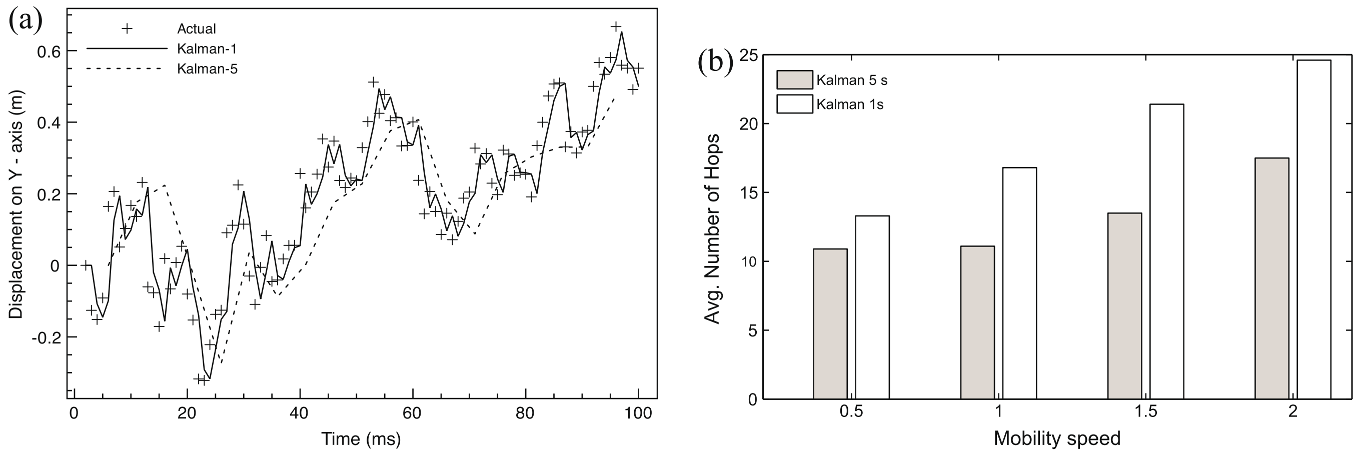

Fig. 12. The accuracy of the location prediction of the destination over time is shown in (a) while the impact of Kalman filter update periods on route connectivity shown in (b).

the SEARCH protocol improves the PDR performance for both increasing PU ON times as well as when the number of PUs in the system is changed from 10 to 20. Fig. 8(c) provides an insight of the system behaviour in a sample scenario when a single PU is considered on a channel used by both GPSR and a portion of the route in the SEARCH protocol. Here, the end-to-end delay is shown as a function of the simulation time. At 50 and 150 s into the simulation time, a PU arrives in the network, and departs at 100 and 200 s, respectively. The resulting interference due to the PU activity causes high fluctuation of the end-to-end delay on the GPSR [4] protocol. The SEARCH protocol, in the route maintenance stage, immediately identifies the PU occurrence and explores a detour path for the duration of the PU activity, as described in Section 5.1. In this case, the new path latency was acceptable at the destination (latency threshold $L_{Th} = 0.2$ s) and the RERR message was not sent for a fresh exploration. However, SEARCH does not revert back to the optimal path once the PU leaves, thus incurring an acceptable but permanent degradation compared to the initial performance.

6.4.2. Node mobility

We first investigate the accuracy with which the destination is able to predict its future location based on the Kalman filter [20]. In Fig. 12(a), we show the true displacement of the destination node along the *y* axis as against the predicted values, with the displacements being measured from the last reported location. The predicted values track the true displacements closely when the Kalman updates happen every 1 ms and suffer from a large deviation for 5 ms.

We study the effect of the route extension due to movement of the destination and the effectiveness of the Kalman filter based location update function in Fig. 12(b). This supports the protocol description in Section 5.2. After the initial route is formed, we increase the mobility of the destination while keeping the other nodes of the route fixed. The average hop length is computed at the end of each trial running for 100 s and the threshold increase in the number of hops before an RERR is sent is set at 10. We consider two cases, when the Kalman filter based location updates are issued at 1 and 5 s by the destination, respectively. We find that the hop length of the route increases as higher node mobility that proves the prediction algorithm and subsequent route extension work effectively. Moreover, the more frequent the location updates, the more accurately the destination is tracked by the penultimate node. The 1 s updates result in longer routes as compared to the more infrequent update duration of 5 s.

In Fig. 13, we evaluate the performance of the recovery scheme in presence of the mobility of the intermediate nodes of the path, as described in Section 5.1 by keeping a constant PU coverage region and the end-points stationary. The number of generated RREQ messages signaling the need for a fresh route formation is plotted against increasing mobility. When the route maintenance (RM) is switched off, the resulting link failure results in the need of a fresh route formation from the source. When RM is on, SEARCH attempts to find another node that is close to the corresponding anchor point, if the current node moves out of range. We observe that this switching of the node in the route manages keeps the route connected through the PU-free regions, leading to much fewer route outages.

7. Conclusions

In this paper, we presented SEARCH, a distributed routing protocol for mobile CR networks. Our approach jointly optimizes the path and channel decisions so that the end-to-end path latency is minimized. It is sensitive to the PR activity and ensures that the performance of the CR network is minimally affected as well as no interference is caused to the licensed users during their transmission. The route management functionality effectively manages to meet the challenges of a mobile environment and proactively takes corrective actions based on predicted location values.

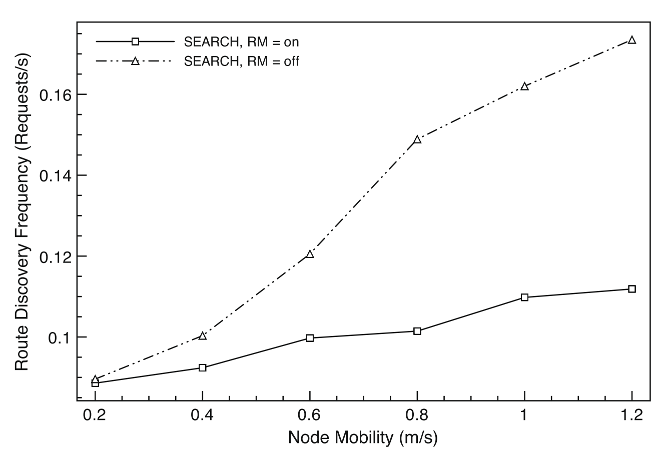

Fig. 13. The reduction in the number of fresh routing requests in the presence of node mobility.

SEARCH can be further enhanced by incorporating a learning based approach that identifies the *type* of the PU, its duty cycle and times of operation. In addition a variety of other channel quality metrics including external interference measurements may be integrated in the next hop selection scheme.

Acknowledgements

The authors would like to thank Prof. Ian Akyildiz for his valuable advice during the course of this work.

Appendix A. Discussion on the path and channel decisions

In this section, we analyze the difference in the costs of circumventing a PU affected region as against changing the transmission channel under *ideal* circumstances. By ideal, we imply that we consider (i) a single PU, (ii) the route detour follows the circumference of the PU region and (iii) the path to the destination, after the channel change, is the shortest path along a straight line. This analysis gives an idea about the likely path and channel choices that will be made by the SEARCH protocol before it is deployed in the network and can be easily extended for other cases.

Consider the PU at location O with a coverage range of R_P^k on channel k. The CR user has a transmission range of R_T and we consider two locations for the decision point (DP), at A and B, respectively. We assume the case for which the route detour is maximum, i.e., the DP, the PU location and the destination D collinear. The destination is at a distance of l_D from the PU. The lines from the AX and DZ form tangents to the circular PU affected region, giving an interior angle $90-\delta$ and $90-\phi$, respectively.

We recall that SEARCH marks a forwarding node as a DP when there exists no node within its focus region that is free of PU activity. Thus, in the best case, the DP is located at A with the next hop neighbor at location B just within the coverage region of the PU. The path detour $\mho_A$ in this case is,

$$\mho_A = \ell_{AX} + \ell_{XZ} + \ell_{ZD} \tag{10}$$

In the worst case, the DP is identified late, when the CR user that forwards the packet is located at B. The detour,with respect to the original location A, $\mho_B$, given as the sum of the length of the arc ℓ_{BXYZ} and the tangent ℓ_{ZD} to the destination,

$$\begin{aligned}\mho_B &= \ell_{AB} + \ell_{BYZ} + \ell_{ZD}\\ \mho_B &= \ell_{AB} + \ell_{BY} + \ell_{YZ} + \ell_{ZD}\end{aligned} \tag{11}$$

Considering $\triangle$AOX, $\delta = \cos^{-1}\left(\frac{R_P^k}{R_P^k+R_T}\right)$. Similarly, in $\triangle$DOZ, $\phi = \cos^{-1}\left(\frac{R_P^k}{l_D}\right)$. From geometrical considerations in Fig. 14,

$$\begin{aligned}\ell_{AB} &= R_T\\ \ell_{AD} &= l_D + R_T + R_P^k\\ \ell_{AX} &= \left\{\left(R_T + R_P^k\right)^2 - \left(R_P^k\right)^2\right\}^{\frac{1}{2}}\\ \ell_{ZD} &= \left\{(l_D)^2 - \left(R_P^k\right)^2\right\}^{\frac{1}{2}}\\ \ell_{XY} &= \left(\frac{\pi}{2} - \delta\right) \times R_P^k\\ \ell_{YZ} &= \left(\frac{\pi}{2} - \phi\right) \times R_P^k\end{aligned} \tag{12}$$

Substituting the terms form Eq. (12) in Eqs. (11) and (10),

$$\begin{aligned}\mho_A &= \left\{\left(R_T + R_P^k\right)^2 - \left(R_P^k\right)^2\right\}^{\frac{1}{2}} + \left(\frac{\pi}{2} - \delta\right)\cdot R_P^k\\ &\quad + \left(\frac{\pi}{2} - \phi\right)\cdot R_P^k + \left\{(l_D)^2 - (R_P^k)^2\right\}^{\frac{1}{2}}\\ \mho_B &= R_T + \frac{\pi \cdot R_P^k}{2} + \left\{(l_D)^2 - (R_P^k)^2\right\}^{\frac{1}{2}}\end{aligned} \tag{13}$$

Along similar lines, the straight line distance from the DP A to the destination D is, $\ell_{ZD} = R_T + \cdot R_P^k + l_D$. Consider a channel q, which is free from all PU activity, and in which such a straight traversal to the destination is possible. The time latencies on the channels with the detour (L_k) and the straight line path (L_q), considering the DP at B and t_s as the channel switching time, are,

$$\begin{aligned}L_q &= t_s + \frac{\left(R_T + \cdot R_P^k + l_D\right)\cdot T_R^k}{R_T}\\ L_k &= \frac{\mho_B \cdot T_R^k}{R_T}\end{aligned} \tag{14}$$

Under ideal conditions, the points A and B mark the minimum and maximum detours, respectively. Thus, typically, the detour may begin from an intermediate point on the segment AB. In the worst case, i.e., at point B, the channel is switched only if the condition $L_q < L_k$ is true. If t_s is large, depending upon the hardware specifications, the detour path on the same channel will always be preferred. Before the network is set in operation, the condition in Eq. (14) for the maximum detour may be checked to estimate if the SEARCH protocol will satisfactorily find the path-channel optimal routes, under the operating limits of the network hardware.

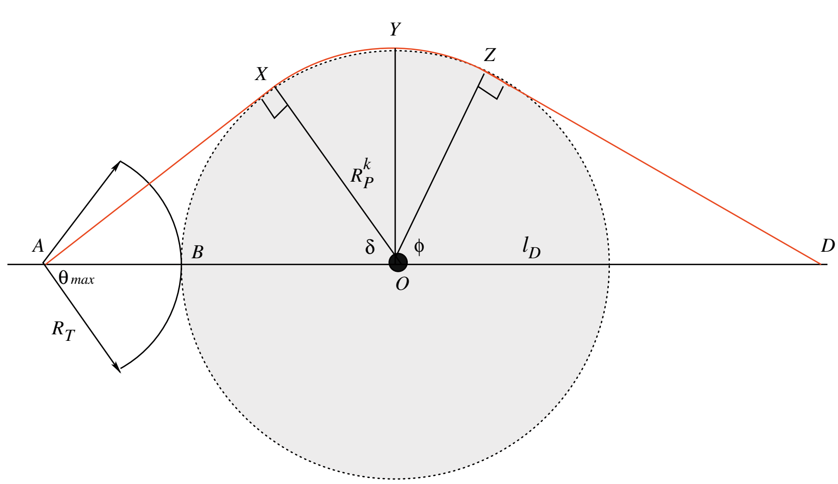

Fig. 14. Circumventing the PU region and the associated path detour.

References

[1] I.F. Akyildiz, W.Y. Lee, M.C. Vuran, S. Mohanty, Next generation/dynamic spectrum access/cognitive radio wireless networks: a survey, Elsevier Comput. Netw. J. 50 (2006) 2127–2159.

[2] S. Krishnamurthy, M. Thoppian, S. Venkatesan, R. Prakash, Spectrum aware on-demand routing in cognitive radio networks, in: Proc. of IEEE Milcom, October 2005, pp. 455–460.

[3] G. Cheng, W. Liu, Y. Li, W. Cheng, Spectrum aware on-demand routing in cognitive radio networks, in: Proc. of IEEE DySPAN, April 2007, pp. 571–574.

[4] B. Karp, H.T. Kung, GPSR: greedy perimeter stateless routing for wireless networks, in: Proc. of ACM MobiCom, August 2000.

[5] B. Leong, B. Liskov, R. Morris, Geographic routing without planarization, in: Proc. of Symp. on Network Sys. Design and Implementation (NSDI 2006), San Jose, CA, May 2006.

[6] C. Xin, B. Xie, C. Shen, A novel layered graph model for topology formation and routing in dynamic spectrum access networks, in: Proc. of IEEE DySPAN, April 2007, pp. 308–317.

[7] Q. Wang, H. Zheng, Route and spectrum selection in dynamic spectrum networks, in: Proc. of IEEE Consumer Comm. and Networking Conf. (CNCC), January 2006, pp. 625–629.

[8] V. Dumitrescu, J. Guo, Context assisted routing protocols for inter-vehicle wireless communication, in: Proc. of IEEE Intelligent Vehicles Symposium, June 2005, pp. 594–600.

[9] K.C. Lee, J. Haerri, U. Lee, M. Gerla, Enhanced perimeter routing for geographic forwarding protocols in urban vehicular scenarios, in: Proc. of IEEE Workshop on Automotive Networking and Applications, in Conjunction with IEEE Globecom, November 2007, pp. 26–30.

[10] C.H. Chou, K.-F. Ssu, H.C. Jiau, Geographic forwarding with dead-end reduction in mobile ad hoc networks, IEEE Trans. Veh. Technol. 57 (4) (2008) 2375–2386.

[11] D. Son, A. Helmy, B. Krishnamachari, The effect of mobility-induced location errors on geographic routing in mobile ad hoc sensor networks: analysis and improvement using mobility prediction, IEEE Trans. Mob. Comput. 3 (3) (2004) 233–245.

[12] R. Pal, Efficient routing algorithms for multi-channel dynamic spectrum access networks, in: Proc. of IEEE DySPAN, April 2007, pp. 288–291.

[13] IEEE Std 802.11b-1999/Cor 1-2001, 2001.

[15] Multi-channel multi-interface simulation in NS2. Available from: <http://www.cse.msu.edu/wangbol/ns2/nshowto8.html>.

[16] M. Alicherry, R. Bhatia, L.E. Li, Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks, in: Proc. of ACM MobiCom, August 2005, pp. 58–72.

[17] J. So, N.H. Vaidya, Multi-channel Mac for ad hoc networks: handling multi-channel hidden terminals using a single transceiver, in: Proc. of ACM MobiHoc, May 2004, pp. 58–72.

[18] T. Luo, M. Motani, V. Srinivasan, CAM-MAC: a cooperative asynchronous multi-channel MAC protocol for ad hoc networks, in: Proc. of the Intl. Conf. on Broadband, Comm. and Systems (BROADNETS), October 2007, pp. 1–10.

[19] M. Rudafshani, S. Datta, Localization in wireless sensor networks, in: Proc. of the Intl. Symp. on Information Proc. in Sensor Net. (IPSN), April 2007, pp. 51–60.

[20] S.M. Kay, Fundamentals of Statistical Signal Processing: Estimation Theory, Prentice-Hall, Inc., 1993, ISBN: 0-13-345711-7.

[21] C. Chereddi, P. Kyasanur, N.H. Vaidya, Design and implementation of a multi-channel multi-interface network, in: ACM REALMAN Workshop, May 2006.