High Throughput Spectrum-aware Routing for Cognitive Radio Networks

Ashwin Sampath, Lei Yang, Lili Cao, Haitao Zheng, Ben Y. Zhao
Computer Science Department, University of California at Santa Barbara
{*ashwins, leiyang, lilicao, htzheng, ravenben*}*@cs.ucsb.edu*

***Abstract*—Dynamic spectrum networks enable fast deployment of new wireless technologies by effectively utilizing allocated yet unused wireless spectrum. By sensing and utilizing available wireless channels, cognitive radio devices can provide high throughput, low latency communication. Existing schemes for channel assignment suffer drawbacks in throughput and reachability in the presence of location-dependent channel availability. We propose *SPEctrum-Aware Routing Protocol (SPEAR)*, a robust and efficient distributed channel assignment and routing protocol for dynamic spectrum networks based on two principles: integrated spectrum and route discovery for robust multi-hop path formation, and distributed path reservations to minimize inter- and intra-flow interference. Through simulations and testbed measurements, we show that SPEAR establishes robust paths in diverse spectrum conditions and provides near-optimal throughput and end-to-end packet delivery latency. SPEAR performs extremely fast flow setup and teardowns, and can maintain interference-free flows in the presence of variance in channel availability.**

I. Introduction

Dynamic spectrum networks enable fast deployment of new wireless technologies by effectively utilizing allocated yet unused wireless spectrum. In this model, wireless nodes equipped with *cognitive radios* [11] do not operate on statically assigned spectrum. Instead, they identify and use locally unused licensed spectrum while avoiding disruptions to legacy users who own the spectrum, *e.g.* analog TV broadcast stations.

Such flexibility means cognitive radios can obtain spectrum within a wider range of channels and adapt to spectrum variations, making them ideal for maintaining robust communications in diverse environments. For example, first responders can use cognitive radios to stream video and audio during disaster recovery operations despite interference from local microwave or radio emissions.

We consider the problem of multi-hop routing in an *ad-hoc* network using cognitive radios. Our goal is to provide persistent, high-throughput communication by optimally selecting paths and channel assignments. While similar problems have been explored in conventional multi-channel systems, multi-hop routing in dynamic spectrum systems faces a fundamentally new challenge because of "heterogeneous spectrum availability." Devices in dynamic spectrum systems must yield to nearby legacy users, *i.e.*, they cannot use any spectrum band occupied by a legacy user. Therefore, the set of spectrum bands a device can legally use, hereby referred to as "spectrum availability," depends on its physical location and can vary (slowly) over time.

The introduction of heterogeneous spectrum availability creates a new research challenge: *how to establish and maintain reliable high-throughput communication across regions of different spectrum availability*? And in the context of an ad-hoc network with dynamic flows, how to achieve this using a computationally efficient distributed solution?

We first consider the feasibility of applying prior work on distributed approaches for single-radio multi-channel routing and channel assignment. Existing work can be classified by the granularity of the channel assignment decision: *per-packet* assignment, *per-link* assignment [14], *flow-based* [8] and *component-based* channel assignment [15]. Both flow- and component-based approaches focus on low-complexity end-to-end channel assignment by assigning one channel per flow. However, under spectrum heterogeneity, source and destination pairs often do not share any common available channels. Hence, these approaches will inevitably reject many flows. On the other hand, link-based assignment such as MMAC [14] can potentially address spectrum heterogeneity by allowing links on each flow to use different channels. However, this approach is known to be flow-unaware and cannot optimize end-to-end performance [15]. In particular, links on the same flow compete for channels on a per-packet basis, significantly degrading end-to-end throughput.

We propose *SPEAR (SPEctrum-Aware Routing)*, a routing protocol that supports high-throughput packet transmission in the presence of spectrum heterogeneity. To achieve persistent end-to-end performance, our aim is to integrate the end-to-end optimization of flow-based approaches, with the flexibility of link based approaches to address spectrum heterogeneity. Briefly, SPEAR integrates spectrum discovery with route discovery to cope with spectrum heterogeneity, and let nodes coordinate to assign channels on a per-flow basis to minimize interference and attain near-optimal throughput. Both path discovery and channel coordination are distributed and incur low computational and communication complexity.

II. Background and Related Work

Our problem context is a multi-hop, ad hoc, dynamic spectrum network. Next we briefly overview dynamic spectrum networks, focusing on unique properties that distinguishes them from conventional multi-channel networks.

A. Dynamic Spectrum Networks

There are two types of devices in a dynamic spectrum network. *Primary devices* are legacy wireless devices who

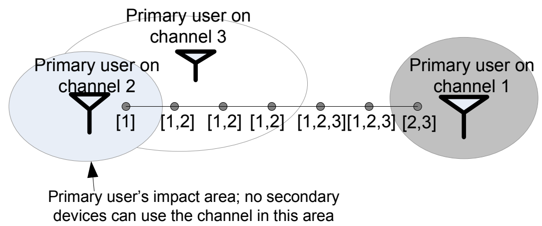

Fig. 1. An example of dynamic spectrum availability, marked as [n..], depends on the activity of primary users.

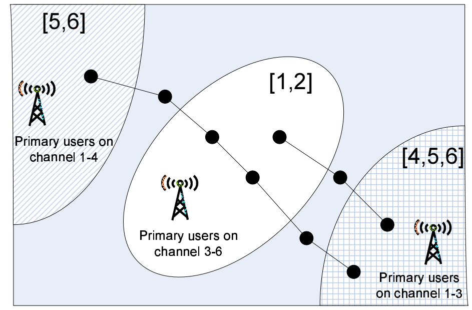

Fig. 2. Routing across regions with heterogeneous spectrum availability.The introduction of primary users creates three islands of different spectrum availability, marked by [i_1,...,i_2]. Nodes outside these islands have all channels (1-6) available.

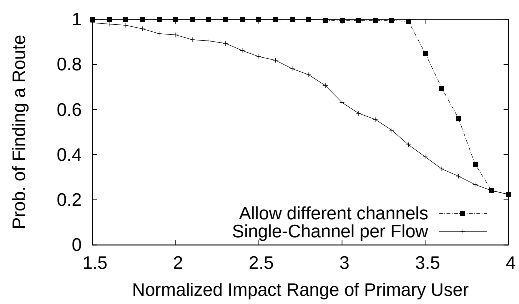

Fig. 3. The probability of finding a route in dynamic spectrum systems; with and without uniform per-flow channel assignment.

own licensed spectrum but do not fully utilize it. Examples include UHF TV and public safety broadcast stations. *Secondary devices* are next-generation cognitive radio devices who opportunistically exploit locally unused licensed spectrum without disrupting operations of primary devices. Secondary devices that detect the presence of a primary devices on a channel must switch to other channels.

Figure 1 illustrates a sample dynamic spectrum system, where two secondary users communicate in the presence of 3 primary users. No single channel is available across the entire path, and the two endpoints must use assign different channels to each link to avoid disrupting nearby primary users.

B. Related Work

There are two network architecture for managing spectrum usage: a *centralized* architecture through a central broker or a *distributed* architecture through node coordination or contention. An overview of existing solutions can be found in [1] and its references. Our work differs from these works in that we consider multi-hop ad hoc routing scenarios while most existing works consider single-hop or cellular architectures.

Our work shares similarities with the routing and channel assignment problem in multi-channel networks. There are numerous work in this area, including centralized [2], topology-optimized [13] and multi-radio based [10], [7] solutions. However, they are not applicable to our problem, because our system is ad hoc with arbitrary topology and our problem falls into *distributed* solutions with *single-radio* for data communication. In Section III, we will take a closer look at existing approaches and evaluate their applicability to multi-hop routing in dynamic spectrum networks.

III. Routing in Dynamic Spectrum Networks

Routing protocols for dynamic spectrum networks should exploit the flexibility and power of cognitive radios while addressing unique challenges not present in traditional networks. Next, we discuss the challenges on routing in dynamic spectrum networks and consider the feasibility of applying approaches previously proposed for conventional networks.

A. The Impact of Heterogeneous Spectrum Availability

Because secondary devices must yield to primary users, their available spectrum is location-dependent and hence heterogeneous across the network. In the example in Figure 2, the introduction of primary users has produced three islands with heterogeneous spectrum availability. To set up multi-hop connections between node pairs with different spectrum availability, intermittent bridge nodes have to switch between multiple channels. That is, links on each path need to communicate on different channels.

Using random topologies, we examine the probability of finding a route between any two nodes, if links on each route are restricted to use a single channel, or if they can use different channels. Figure 3 compares the success rate under different impact range values of primary users, normalized over the secondary user's transmission range. We see that having a single-channel per route becomes an exception. To avoid rejecting connections, it is crucial to allow links on each path to user different channels. Finally, when primary user's impact area become fairly large, the usable channels diminish, and only few routes succeed.

B. Examining Existing Approaches

Numerous approaches have been proposed for efficient routing in conventional single-radio multi-channel wireless systems. We examine if and how well each of these approaches addresses our needs in dynamic spectrum networks.

First, flow- or component-based approaches use a single channel per flow or connected component [8], [15]. Figure 3 shows that the presence of heterogeneous spectrum regions often breaks this assumption, resulting in numerous rejected flows. One extension is to use different channels in different sections of the path. But it still suffers from intra-flow interference that significantly reduces achievable throughput. Finally, any changes in spectrum availability will require reconfiguration of the entire flow or connected component, significantly disrupting application traffic.

Second, distributed link-based routing protocols decouple channel assignment from routing by allowing links to choose channels independently to maintain neighbor connectivity [6], [14]. While this enables each node to choose channels based on local availability, it is known to be flow-unaware, and cannot optimize end-to-end multi-hop performance [15]. Existing link-layer approaches use per-link per-timeslot channel assignment [14], which can produce suboptimal channel assignments and control message contention in each time slot.

C. Routing across Heterogeneous Spectrum Regions

After examining current approaches, we conclude that to route across regions of heterogenous spectrum availability, we need a distributed end-to-end approach to optimize route performance, while allowing flexibility in channel usage to cope with spectrum heterogeneity.

We propose *SPEctrum-Aware Routing (SPEAR)*, a new routing protocol for high-throughput multi-hop routing in dynamic spectrum systems. The unique properties of SPEAR include:

- Integrate spectrum discovery with route discovery to cope with spectrum heterogeneity, and obtain optimal usage of available channels.
- Coordinate channel usage explicitly across nodes to optimize channel assignment on a per-flow basis, and to minimize inter-flow interference and interference.
- Exploit local spectrum heterogeneity and assign different channels to links on the same flow to minimize intra-flow interference.

Finally, SPEAR is distributed and incurs low computational and communication complexity. Utilizing spectrum heterogeneity, SPEAR can attain near-optimal end-to-end throughput that does not degrade with additional hops.

IV. SPEAR Protocol Design

In this section we describe in detail the SPEAR protocol. We note that SPEAR assumes each device has one dedicated control radio and one data radio. This is different from the conventional multi-radio devices in mesh networks. Next, we begin with an overview and then details of each component.

SPEAR overview We illustrate the high level operation of SPEAR using an example. As shown in Figure 4, node S seeks a multi-hop interference-free path to destination D. First, S initiates a connection to D by performing *Spectrum-aware Route Discovery*. It broadcasts an AODV-style [12] route discovery message to its neighbors, who forward them on. In addition to locating a forwarding path to D, these messages also accumulate information about each node's available channels (those not used by primary users and not reserved by other flows). Unlike AODV, however, SPEAR allows multiple paths to propagate to the destination. To avoid broadcast congestion, nodes eliminate routing loops and use per-flow state to limit the number of paths discovered.

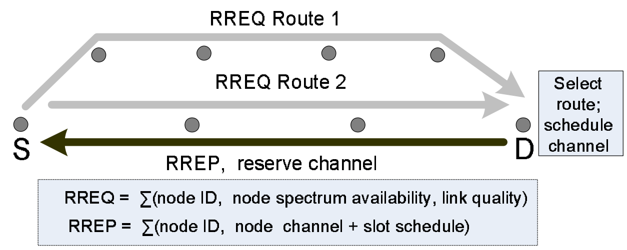

Fig. 4. Establishing a multi-hop virtual circuit between source S and destination D using the SPEAR protocol.

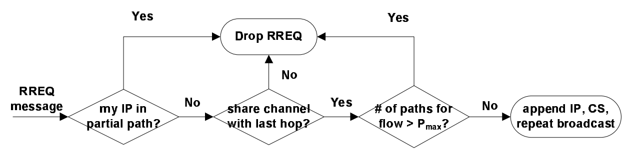

Fig. 5. The flow-chart diagram for a node processing a RREQ message.

Second, the destination D performs *Route Selection and Scheduling* by collecting a bounded number of routes and choosing the optimal route based on maximum end-to-end throughput, hop count, and other potential routing metrics [4]. D then computes the optimal channel assignment for the route, and embeds the assignment and the MAC protocol to use in a route reply message sent back to S. Next, each node along the path performs *Route Setup and Channel Reservation* by parsing the information in the route reply message to determine its channel usage, and forwards an explicit channel reservation message to all neighbors within their interference impact area. Reservation state is maintained as soft state by periodic beacons. Finally, as channel availability changes with the actions of primary users, nodes perform *Local Adaptation* by modifying their local channel usage to maintain flow connectivity. If local adaptations fail, SPEAR invokes routing level route repair mechanisms to restore the path.

We now describe SPEAR components in further detail.

A. Spectrum-aware Route Discovery

To provide robust connectivity in the presence of heterogeneous spectrum availability, SPEAR integrates traditional on-demand route discovery with spectrum sensing. As shown in Figure 1, each node maintains lists of locally available channels (channelSet A) that are not occupied by primary users and that not reserved by nearby neighbors. A SPEAR node S performs spectrum-aware route discovery by broadcasting a RouteRequest (RREQ) message on the dedicated control channel, with its channelSet A_S inside. Each RREQ message is uniquely identified by the source and destination IP addresses.

As node i receives a RREQ message, it first examines the current partial path for its own address. If found, i has forwarded this message before, and now drops it in order to break the routing loop. Otherwise, i checks to see if it shares a common channel with the previous hop node. If so, i appends its identifier and its channelSet A_i to the payload, and broadcasts the message. Otherwise, this link cannot be established, and the RREQ is dropped. We illustrate the steps in RREQ processing in Figure 5.

Minimizing RREQ Traffic. Note that unlike traditional on-demand route discovery protocols such as AODV, SPEAR needs to discover multiple diverse paths to the destination. By default, redundant paths are not suppressed, and all possible paths are sent to the destination for route selection. We use two mechanisms to reduce the amount of RREQ broadcast traffic. First, we can limit the number of routes forwarded by each node using a parameter P_{max}. Each node keeps a per-flow counter, and only forwards the first P_{max} RREQ messages. Second, when the destination node D receives the first RREQ, it starts a per-flow timer T_R. When T_R expires, D selects the optimal route and performs channel assignment based on routes received, and sends a RouteReply (RREP) back to the source. On receiving the RREP, each node reserves the assigned channel and timeslots. All notified nodes will drop additional RREQ messages for this flow.

Intersecting Flows. Since SPEAR uses TDMA-style channel scheduling, a flow obtains maximum throughput when it does not intersect with any other flow. Sharing any node with another flow reduces the end-to-end throughput by half. Under certain topologies however, the only path between two nodes intersects with an existing flow. When a RREQ messages reaches a node i already servicing a flow, it records the number of existing flows on i and their time schedules. The number of existing flows serviced defines a throughput limit for any path crossing i. Crossing one flow intersection means a flow's maximum throughput is limited to half of optimal. Any additional intersections after the first does not incur further throughput degradation, since the flow is already sending at half of the maximum rate. We discuss in Section IV-B how the destination utilizes this information during route selection.

B. Route Selection and Scheduling

During route discovery, multiple RREQ messages are forwarded along different paths towards the destination D. On arrival, each RREQ message encodes a full path from the source. For each node on the path, it provides its identifier (IP), available channelSet, and the time schedule of any flows it is servicing. Upon receiving the first RREQ messages for a given flow, D starts a timer T_R, and collects all RREQ messages until T_R expires. Subsequent RREQ messages for the flow are dropped. In this section, we describe in detail how destination D analyzes its set of received paths to select the best route and compute its optimal channel and time slot assignment.

Route Selection. The destination D can use a variety of policies to determine the most desirable route. By observing the number of flows at any intersections along a route, D can estimate the route's maximum throughput. D can also seek to minimize end-to-end latency by choosing routes with lower hop-count. Finally, we can also consider *quality-based metrics* such as ETX [3], ETT, WCETT [5], by embedding each link's quality into the RREQ message. In general, the destination computes a policy-driven utility value for each candidate route based on a combination of the above factors, and selects the route with the highest utility. In our implementation, we select routes by sorting on maximum possible throughput, using minimal hop-count to break ties.

Channel Assignment and Scheduling. Using information from RREQ messages, D computes a per-hop channel usage schedule for the route it has chosen. By combining the available channel set from link endpoints, each link now has a list of available channels that will not disrupt primary users or interfere with neighboring flows. The propose schedule assigns each link with a channel from its availability list. It also avoids self-interference by assigning orthogonal channels to conflicting links on the flow. Because the data radio is half-duplex, we divide time into equal-sized time slots, with each device alternating between transmit and receive modes on consecutive time slots.

The problem of channel assignment can be reduced into a graph coloring problem by mapping each link into a vertex and its availability list as the color list. If two links conflict then they are connected in the conflict graph. Links that are on even and odd time slots will form two conflict graphs. The optimal assignment is to use the minimum number of colors to color each vertex with a color from its list, so that no two connected vertices have the same color. We use a heuristic-based approximation that colors vertexes iteratively, each time selecting the vertex with the fewest colors available.

Scarce Spectrum Scenarios. In rare circumstances, a concentration of primary users can result in sporadic channel availability insufficient for conflict-free channel/slot assignments. One solution is to divide time into shorter slots to create additional "logical" orthogonal channels, at the cost of finer-grain time synchronization. We propose a simple alternative: allow selective links to share a channel with its conflicting neighbors by using CSMA/CA to avoid self-interference.

Following channel assignment and scheduling, D sends the per-hop channel schedule and protocol setting in a RouteReply (RREP) message along the path back to the source node.

C. Reservation-based Route Management

Nodes use explicit channel reservation messages to coordinate channel usage. This implies that SPEAR nodes need to have the knowledge of conflicting nodes. We noticed that there have been considerable contributions on interference discovery supported by testbed verifications, such as [9]. In addition, measurement results have verified the assumption of 2-hop interference model for indoor networks. Therefore, we assume that cognitive radios can incorporate some of these features with its sensing capability to reliably discover interfering neighbors. In this paper, for simplicity, we assume a 2-hop interference model.

Next, we briefly describe how SPEAR uses soft-state reservation announcements in both route setup and teardown.

Route Setup. If a node is transmitting data, it must broadcast periodic channel reservations that announce its channel use. When a node receives a RREP message, it schedules its channel usage according to the defined schedule, and either modifies its existing reservations or begins a new reservation broadcast (if it had been idle). Each reservation has an implicit timeout period T_L during which it is valid. This soft-state approach ensures simplifies management and provides robustness against node failures and node mobility. To minimize broadcast overhead and contention, each reservation message has a time-to-live (TTL) field that limits its reach to neighbors within the sender's interference range.

Route Teardown. SPEAR handles route teardowns implicitly. When a flow terminates, nodes along the path are notified to stop sending reservation messages. Channels whose reservations have timed out are assumed to be open. If faster channel reuse is desired, nodes along the path can send an explicit teardown message to revoke existing reservations.

V. Simulation Results

We evaluate SPEAR performance using Qualnet simulations on a 1000m x 1000m grid. Unless specified otherwise, we assume a traffic model consisting of unidirectional UDP traffic. Each node is equipped with a single half-duplex cognitive radio for data transmission and a single half-duplex normal radio for control transmission. The available spectrum is divided into 12 channels. Each cognitive radio can access one channel at a time, while primary users can claim multiple channels simultaneously. Both the cognitive and control radios are configured for a data rate of $12Mbps$. In SPEAR, each cognitive radio follows $85ms$ time slots with two $5ms$ guard-band, with channel switching delay of $80us$. The control radio uses the 802.11 CSMA/CA MAC protocol. To simulate spectrum heterogeneity, we use the scenario of Figure 2.

A. SPEAR Protocol Overhead

In SPEAR, on-demand route discovery messages and periodic channel reservation messages are the two main sources of protocol overhead.

First we examine the route discovery overhead in terms of route setup and tear-down delay. We define the route setup delay as the time from when a source broadcasts a RREQ message to when it receives the RREP message; and the route tear down delay as the time required for a tear-down request to propagate through the entire route and to all neighboring nodes within the interference range. In this experiment, we set up flows sequentially one flow every 2 seconds and send routing messages via the control radio. Figure 6 shows both delay measures vs. the number of path hops. We see that SPEAR's setup delay ($< 100ms$) is reasonably small despite the efforts to discover multiple paths. The tear down delay is $< 15ms$. Hence, SPEAR can support short-lived flows, and is extremely responsive to user requests for communication.

While on-demand route discovery results in an initial route setup overhead, channel reservations result in continuous overhead for the entire lifetime of a flow. For a system with 12 channels, a maximum of 30 flows and two channel reservation messages per second, the average control channel bandwidth consumed by SPEAR was measured to be $< 12kbps$. Hence, SPEAR can be successfully deployed even in cognitive radio environments with a narrow band control channel.

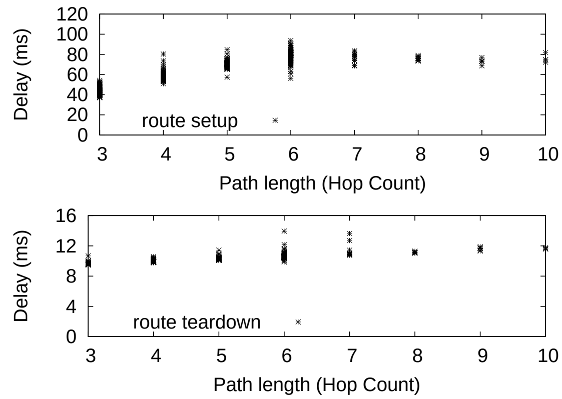

Fig. 6. SPEAR's route setup and tear-down delay

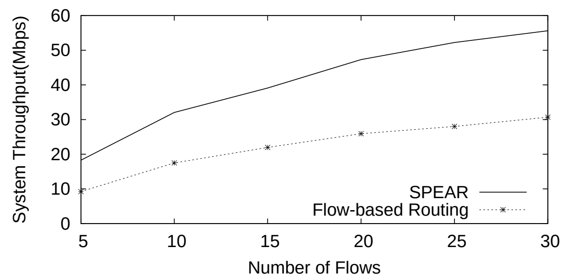

Fig. 7. Aggregate system throughput using 12 channels and no primary users.

B. Aggregated System Throughput

We compare the aggregate system throughput of SPEAR with an optimistic Flow-based scheme where each flow is assigned a channel that does not conflict with neighboring flows.

Homogeneous Spectrum Availability First we examine the aggregate system throughput under homogeneous spectrum conditions without any primary users. This scenario presents a spectrum rich environment where all the channels are available for data communication, which is ideal for evaluating the peak performance of SPEAR and flow-based scheme.

Figure 7 plots the aggregate system throughput of SPEAR and Flow-based scheme as a function of the number of flows in the system. SPEAR achieves a two fold gain over Flow-based routing for both heavy and light traffic conditions. We observe that SPEAR achieves almost twice the performance of Flow-based scheme. This is primarily because SPEAR eliminates intra flow interference using a non-conflicting channel assignment on each path, resulting in higher per flow throughput.

Heterogeneity Spectrum Availability We now examine SPEAR's throughput performance in the presence of primary users. Using the scenario in Figure 2 we place two primary users at diagonally opposite corners of the grid, and a third primary user at the center of the grid. While the primary users at the diagonally opposite corners occupy channels 1 to 6, the primary user at the center of the grid occupies channels 5 to 12. The primary user impact range is varied as a multiple of the secondary user transmission range.

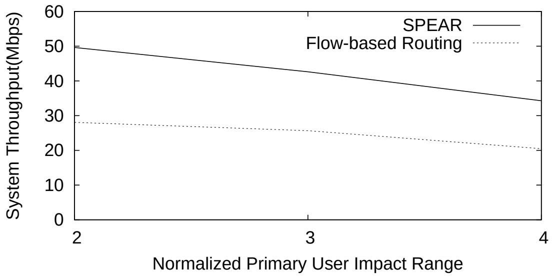

Fig. 8. Aggregate system throughput using 12 channels and 3 primary users, assuming 30 random flows.

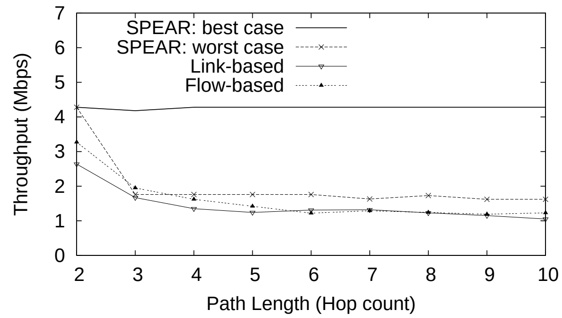

Fig. 9. Throughput on linear topology.

For 30 randomly chosen source-destination pairs, Figure 8 shows the aggregate system throughput of SPEAR and the Flow-based scheme as the normalized primary user impact range is increased. We see that even when primary users impact a large part of the network, SPEAR achieves significant improvement over Flow-based approach. This is because the following two reasons. First, due to the spectrum aware property, the success rate of end to end path discovery is significantly higher in SPEAR than in the Flow-based scheme. Second, while intra-flow interference limits per-flow throughput in the Flow-based scheme, SPEAR maintains a higher throughput by eliminating intra-flow interference.

C. SPEAR vs. Link-based Approach

We compare SPEAR with the link-based approach using an approximation of MMAC [14]. In this experiment, we use a simple linear topology *favorable* to the MMAC protocol. In a random topology, neighbor links would compete for links, producing large control contention.

We compare the best and worst case performance of SPEAR with MMAC. While in the best case, SPEAR uses a minimum of 2 channels to perform a non-conflicting channel assignment for the path, in the worst case all the links are forced on to a single channel due to the availability of a single common channel at each node. We use the even-odd time slot structure for both the best and worst case evaluation. The MMAC system is configured to use all the 12 channels. As a reference, we also plot the throughput of the flow-based approach.

Figure 9 illustrates the average flow throughput performance. In the best case, SPEAR's flow throughput is constant regardless of the hop count. In the worst case, however, the throughput decreases initially and then remains constant. This degradation in throughput is due to the contention among links in paths of length greater than 2. While in the best case SPEAR achieves more than 180% improvement over the link-based approach, even in the worst case SPEAR performs marginally (35%) better than the link-based and the flow-based approach.

VI. Conclusion

Our work builds a foundation for dynamic spectrum networks to support high-throughput multi-hop routing under spectrum heterogeneity. SPEAR combines two simple yet powerful features: integration of spectrum and route discovery to establish communications across areas of varying spectrum availability, and distributed path reservation to minimize inter- and intra-flow interference. Extensive simulations confirm the efficiency of SPEAR and demonstrate its capability to provide high-throughput, robust multi-hop communications. SPEAR is ideal for communications under unknown and dynamic spectrum conditions, *i.e.* disaster recovery or military operations.

Acknowledgments

We gratefully acknowledge support from NSF through grants CNS-0721961 and CNS-0546216, and support from DARPA through BAA04-11.

References

[1] Akyildiz, I. F., Lee, W. Y., Vuran, M., and Mohanty, S. NeXt generation/dynamic spectrum access/cognitive radio wireless networks: A survey. *Computer Networks Journal* (2006).

[2] Alicherry, M., Bhatia, R., and Li, L. E. Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks. In *Proc. of MobiCom* (August 2005).

[3] De Couto, D. S. J., et al. A high-throughput path metric for multi-hop wireless routing. In *Proc. of MobiCom* (Sept. 2003).

[4] Draves, R., Padhye, J., and Zill, B. A high-throughput path metric for multi-hop wireless routing. In *Proc. of SIGCOMM* (August 2004).

[5] Draves, R., Padhye, J., and Zill, B. Routing in multi-radio,multi-hop wireless mesh networks. In *Proc. of MobiCom* (September 2004).

[6] Kyasanur, P., et al. Multi-channel mesh networks: Challenges and protocols. *IEEE Wireless Communications* (April 2006).

[7] Kyasanur, P., and Vaidya, N. H. Routing and interface assignment in multi-channel multi-interface wireless networks. In *WCNC* (2005).

[8] Kyasanur, P., and Vaidya, N. H. A routing protocol for utilizing multiple channels in multi-hop wireless networks with a single tranceiver. In *Proc. of Qshine* (August 2005).

[9] Lim, H., Lim, C., and Hou, J. A coordinate-based approach for exploiting temporal-spatial diversity in wireless mesh networks. In *Proc. of MobiCom* (September 2006).

[10] Lin, X., and Rasool, S. A distributed joint channel-assignment, scheduling and routing algorithm for multi-channel ad hoc wireless networks. In *Proc. of INFOCOM* (2007).

[11] Mitola III, J., and Maguire Jr., G. Q. Cognitive radio: making software radios more personal. *IEEE Personal Communications 6*, 4 (Aug. 1999), 13–18.

[12] Perkins, C., and Royer, E. Ad hoc on-demand distance vector routing. In *Proc. of WMCSA* (February 1999), ACM.

[13] Raniwala, A., and Chiueh, T. Architecture and algorithms for an ieee 802.11-based multi-channel wireless mesh network. In *Proc. of INFOCOM* (2005).

[14] So, J., and Vaidya, N. Multi-channel MAC for ad hoc networks: Handling multi-channel hidden terminals using a single transceiver. In *Proc. of MobiHoc* (May 2004), ACM.

[15] Vadentham, R., et al. Component based channel assignment in single radio, multi-channel ad hoc networks. In *Proc. of MobiCom* (September 2006).