IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 19, 2016, accepted September 7, 2016, date of publication September 23, 2016,
date of current version October 31, 2016.

Digital Object Identifier 10.1109/ACCESS.2016.2613122

Route Selection for Multi-Hop Cognitive Radio Networks Using Reinforcement Learning: An Experimental Study

AQEEL RAZA SYED[1], KOK-LIM ALVIN YAU[1], JUNAID QADIR[2], (Senior Member, IEEE), HAFIZAL MOHAMAD[3], NORDIN RAMLI[3], AND SYE LOONG KEOH[4]

[1]Faculty of Science and Technology, Sunway University, 46150, Selangor, Malaysia
[2]Information Technology University, 54000, Lahore, Pakistan
[3]Wireless Network and Protocol Research Lab, MIMOS Berhad, 57000, Kuala Lumpur, Malaysia
[4]School of Computing Science, University of Glasgow Singapore, 737729 Singapore

Corresponding author: A. R. Syed (12010591@imail.sunway.edu.my)

This work was supported in part by the Malaysian Ministry of Science, Technology and Innovation under Science Fund under Grant 01-02-16-SF0027, in part by the Malaysian Ministry of Education Fundamental Research Grant Scheme under Grant FRGS/1/2014/ICT03/SYUC/02/2, and in part by the Small Grant Scheme (Sunway-Lancaster) under Grant SGSSL-FST-CSNS-0114-05 and Grant PVM1204.

ABSTRACT Cognitive radio (CR) enables unlicensed users to explore and exploit underutilized licensed channels (or white spaces). While multi-hop CR network has drawn significant research interest in recent years, majority work has been validated through simulation. A key challenge in multi-hop CR network is to select a route with high quality of service (QoS) and lesser number of route breakages. In this paper, we propose three route selection schemes to enhance the network performance of CR networks, and investigate them using a real testbed environment, which consists of universal software radio peripheral and GNU radio units. Two schemes are based on reinforcement learning (RL), while a scheme is based on spectrum leasing (SL). RL is an artificial intelligence technique, whereas SL is a new paradigm that allows communication between licensed and unlicensed users in CR networks. We compare the route selection schemes with an existing route selection scheme in the literature, called highest-channel (HC), in a multi-hop CR network. With respect to the QoS parameters (i.e., throughput, packet delivery ratio, and the number of route breakages), the experimental results show that RL approaches achieve a better performance in comparison with the HC approach, and also achieve close to the performance achieved by the SL approach.

INDEX TERMS Cognitive radio, multi-hop network, route selection, reinforcement learning, spectrum leasing.

I. INTRODUCTION

Cognitive radio (CR) is the next-generation wireless communication system that enables unlicensed users (or secondary users, SUs) to explore and exploit underutilized licensed channels (or white spaces), which are owned by the licensed users (or primary users, PUs), in the spectrum. The PUs have exclusive rights to access their respective licensed channels, and the SUs are unaware of their usage patterns unless there are explicit communications between the PUs and the SUs. Two major objectives of CR are to maximize spectrum utilization, and to improve quality of service (QoS), which can be achieved by incorporating cognition (or intelligence) into SUs.

Majority of the research related to CR networks has been limited to theoretical framework [1], [2], and simulation studies [3]–[5]. In recent years, some essential CR functions, such as channel sensing, have been implemented on real testbeds focusing on PHY and MAC layers [6]–[10]. However, there is only perfunctory effort to investigate the network layer through real testbed implementations, and there are three main limitations. Firstly, only a few nodes have been utilized in existing network-layer implementations [12]–[14]. Secondly, monetary constraint has discouraged network-layer implementation as more nodes and computing resources are needed to investigate multi-hop transmission. Thirdly, the underlying layers (i.e., physical and data link layers),

particularly the hardware and software processing delays, can affect the network layer performance [11]–[13], [15]. This means that the choice of hardware and software for the underlying physical and data link layers can significantly affect the network-layer performance, which is undesirable. We address the limitations associated with the network-layer implementation using a simplified system architecture to construct a larger network consisting up to ten nodes in which the universal software radio peripheral (USRP) hosts are directly connected to a single computer using an Ethernet switch. This allows the extension of existing implementations [12], [13] by increasing the number of USRP hosts in the platform. The Ethernet switch reduces the effects of latency so that the performance of network-layer schemes can be analyzed.

In this article, an experimental setup has been deployed to examine route selection schemes in multi-hop CR platform using USRP [16], [17], and GNU radio toolkit [18]. Generally speaking, USRP, which is an off-the-shelf wireless host, enables each SU to autonomously and dynamically configure various operating parameters, such as channel frequency and modulation scheme, for data transmission using GNU radio. GNU radio, which is an open source software platform, generates signals for USRP nodes and performs waveform-specific processes including modulation (e.g., GMSK), as well as packet encoding and decoding. We deploy three route selection schemes based on: 1) the traditional reinforcement learning (RL) approach, 2) a RL approach with average Q-value, and 3) a spectrum leasing (SL) approach. RL is an artificial intelligence technique in which a decision maker (or an agent) learns about its operating environment and makes decisions on action selection that provides system performance enhancement without using prior or explicit knowledge. SL is a new paradigm that allows communication among PUs and SUs in CR networks. The proposed schemes select the best possible route from a SU source node to a SU destination node in a multi-hop CR network in order to improve QoS parameters (i.e., throughput and packet delivery ratio) and routing stability (e.g., the number of route breakages).

Our contribution is to propose and implement three route selection schemes based on RL and SL on real testbed environment using USRP/GNU radio platform with the objective of improving QoS performance in multi-hop CR networks, while taking into consideration the limitations of the underlying USRP/GNU radio platform for network-layer implementation. To the best of our knowledge, this is the first testbed implementation of RL-based and SL-based route selection schemes in CR networks, taking into consideration the limitations of network-layer implementation.

A summary of the notations used in this article is shown in Table 1. The rest of this article is organized as follows. Section II presents related work. Section III presents system architecture. Section IV presents route selection. Section V presents experiment and evaluation. Finally, Section VI presents conclusion and future work.

II. RELATED WORK

In the network layer of CR networks, there are two widely adopted network architectures, namely distributed and centralized models. In the distributed model, each SU node has local spectrum knowledge, which represents the local availability of various channels over time and space [19]. In the centralized model, there is a centralized entity, such as a SU base station, which has full spectrum knowledge in the form of spectrum occupancy map, representing the network-wide availability of various channels over time and space. The SU base station sends updates on the spectrum occupancy map to the rest of the SUs [20]. In this work, both centralized and distributed models are considered. In the centralized model, a SL approach is adopted in which the PUs share their spectrum occupancy map with SUs [21], [22]. In the distributed model, PUs do not share the spectrum occupancy map with SUs, and so the SUs must sense for available channels and infer their available time.

Tremendous work has been done to investigate route selection in centralized and distributed models in CR networks using simulation tools (e.g., Qualnet and NS2) [23]–[26]. However, there has only limited work on route selection conducted on real testbeds (e.g., USRP/GNU radio) [12], [13]; and hence, this is the focus of this article. Various route selection approaches have been proposed with the objectives of maximizing throughput [27], [28], minimizing end-to-end delay [29], [30], maximizing route stability [13], [31], and minimizing route recovery/maintenance cost [32].

Three main spectrum-aware route selection schemes that adopt the distributed model have been implemented on a CR testbed. Firstly, in [14], Nagaraju et al. (2010) implemented a joint cross-layer routing and channel selection scheme on a testbed comprised of three USRP SU nodes to select a next-hop node in a single-hop CR network. Generally speaking, in a single-hop network, a single source node selects one of the two destination nodes based on throughput performance. There are two possible routes in the network. The route selection scheme uses the signal-to-interference-noise ratio, which is dependent on binary phase shift keying and quadrature phase shift keying modulation schemes, to achieve the objectives of maximizing throughput and minimizing the interference with neighboring PUs and SUs. Secondly, in [13], Huang et al. (2011) implemented Coolest Path on a testbed comprising six USRP SU nodes to find a stable route traversing across a multi-hop CR network. There are three possible routes in the network. Coolest Path uses channel availability as the routing metric, which is dependent on the number of channel and route switches, as well as the number of route breakages, in order to achieve the objectives of maximizing throughput and minimizing route recovery. Lastly, in [12], Sun et al. (2014) investigated various route selection schemes, particularly SAMER and CRP, on a testbed comprising up to six USRP SU nodes to find a stable route traversing across a multi-hop CR network. There are four possible routes in the network. The mechanism uses the number of channel

TABLE 1. Summary of notations.

Category	Notations	Description
Network	$P = \{1,2,\ldots,\lvert P\rvert\}$	A set of PUs
	$M = \{m_1,\ldots,m_N\}$	A set of USRP SU nodes
	m_1	SU source node
	$X_{h,j_h \in J_h} = \{m_2,\ldots,m_{N-1}\}$	A set of SU intermediate nodes, where $h = \{1,2,\ldots,\lvert H\rvert\}$ is the number of hops from SU source node m_1, and $j_h \in J_h$ is one of the h-hop nodes from the SU source node m_1
	$m_{n_{h,j_h}}$	SU intermediate node with node identification number n_{h,j_h}
	m_N	SU destination node
	$K = \{k_1, k_2, \ldots, k_k, \ldots, k_{\lvert K\rvert}\}$	A set of routes in the network
	$\mathbb{R}_{m_1,m_N}$	Route record list in the RREQ message sent from a SU source node m_1 to SU destination node m_N
Channel	t_s	Channel sensing time window
	$C = \{c_1, c_2, \ldots, c_c, \ldots, c_{\lvert C\rvert}\}$	A set of channels in the network
	$\tau^p_{c_c,ON}$	ON duration of PU $p \in P$ in channel $c_c \in C$
	$\tau^p_{c_c,OFF}$	OFF duration PU $p \in P$ in channel $c_c \in C$
	$\lambda^p_{c_c,ON}$	Average ON time of PU p on its channel c_c
	$\lambda^p_{c_c,OFF}$	Average OFF time of PU p on its channel c_c
	L_{k_k}	Set of links in a route k_k
	$\varphi^{i,j,k_k}_{t,c_c,OFF}$	Estimated average channel available time of channel c_c on link between SU node i and SU node j of route k_k at time t for RL-based scheme
	$\mathbb{e}^{i,j,k_k}_{t,c_c,OFF}$	Exact channel available time of channel c_c on link between SU node i and SU node j of route k_k at time t for SL-based scheme
	$\Gamma^{k_k}_{\beta,t}$	Channel available time of bottleneck link in route k_k at time t
RL	$s_t^{m_N}$	The state represents a potential SU destination node at time t
	$a_t^{m_{n_{1,j_1}}}$	The action represents a neighbor node $m_{n_{1,j_1}}$, which is a single hop away from a source node m_1
	$Q_t^{m_1}(s_t^{m_N}, a_t^{1,j_1})$	The Q-value for a state-action pair $(s_t^{m_N}, a_t^{1,j_1})$ calculated at source node m_1 for route k_k at time t
	α	Learning rate, and its range is $0 \leq \alpha \leq 1$
	a_t^*	The selected optimal action at time t
	$\bar{Q}_t^{m_1}(s_t^{m_N}, a_t^{m_{n_{1,j_1}}})$	The average Q-value is a mean ratio of the sum of all Q-values up to time t

switches and the number of route breakages as the routing metrics, which are dependent on channel availability time, in order to achieve the objectives of maximizing throughput and minimizing route recovery.

We achieve our contribution by proposing and implementing three route selection schemes based on RL and SL in a real multi-hop CR testbed. The RL and SL approaches address the dynamicity of the PUs' activities (or channel availability) and select the best possible route in a multi-hop CR networks in order to improve QoS parameters, particularly throughput and packet delivery ratio, as well as the number of route breakages, which represents the route stability. Using RL, SUs learn about the average channel available time and select a route that maximizes the SUs' network performance. On the other hand, SL allows PUs to communicate with SUs and lease their channels to them. Hence, the SL approach may be more suitable in a centralized network in which the SUs has direct communication with PUs. In general, SL offers two main advantages. Firstly, it improves the SUs' channel utilization and network performance based on the spectrum occupancy map sent by the PUs to SUs. Secondly, it offers remuneration to PUs in terms of monetary gain or performance enhancement (e.g., SUs help PUs to relay packets [20]). Hence, the main difference between RL and SL is that, SUs are not informed of the channel utilization of PUs in RL, and the SUs are informed of such information in SL.

This article implements the RL-based and SL-based route selection schemes in a real testbed environment using USRP/GNU radio platform. This article also proposes a system architecture to address three main limitations associated with network-layer implementation. Firstly, the system architecture can establish a larger network, which is necessary in multi-hop network-layer implementation for a meaningful investigation in contrast to two nodes (i.e., a transmitter and a receiver) in physical-layer implementation and single-hop

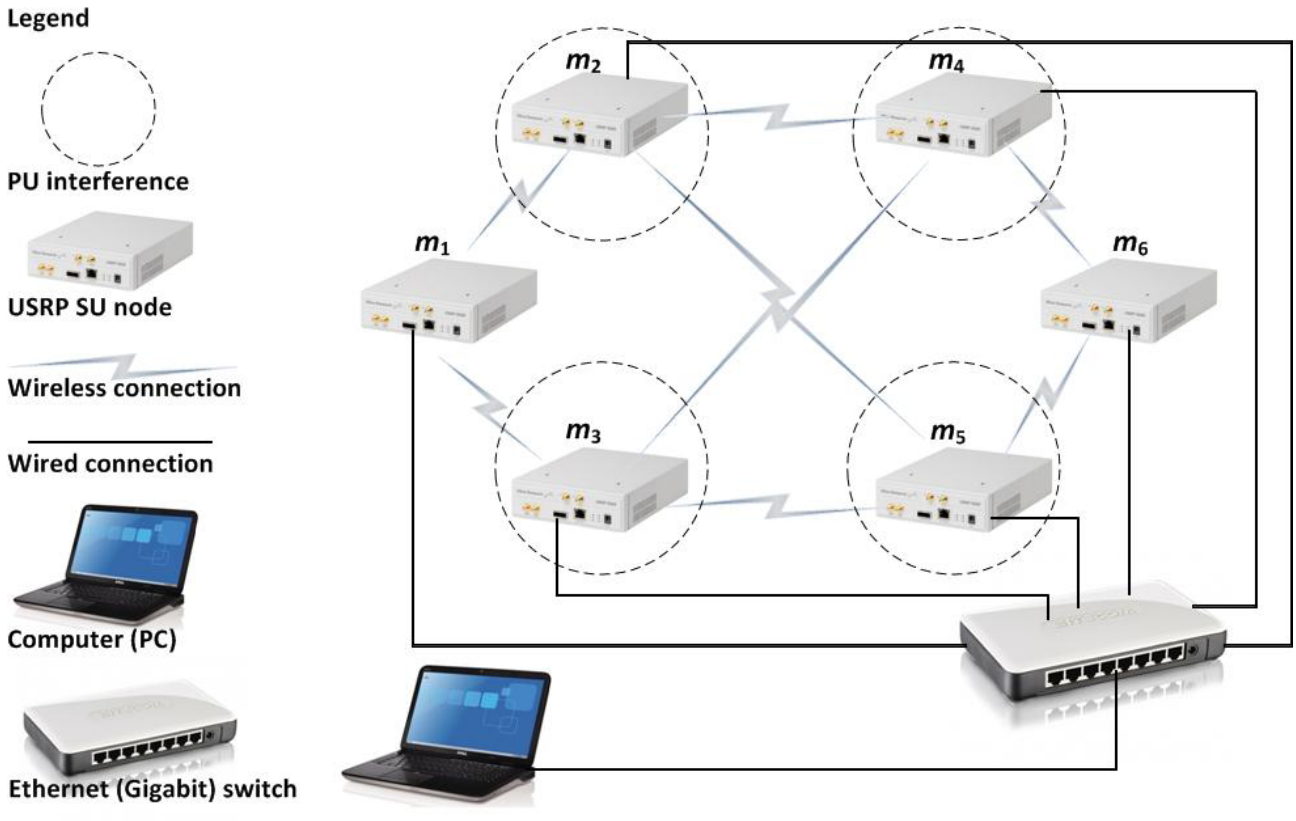

FIGURE 1. A 6-node topology consists of six USRP SU nodes.

transmission (i.e., point-to-point and point-to-multipoint) in data link-layer implementation. While route selection mechanism has been investigated in [14], the implementation focuses on single-hop transmission. Secondly, the system architecture addresses the monetary constraint. The requirement to purchase more equipment (i.e., nodes and computing resources) has discouraged researchers to setup a real testbed environment to investigate the network layer. As an example, in [12], each of the six USRP SU nodes must be connected to a single computer. Our system architecture uses a switch to connect the USRPs to a single computer (addressing the second limitation), and more USRPs can be connected to the switch to establish a larger network (addressing the first limitation). Thirdly, the system architecture reduces the effects of hardware and software processing delays in USRP/GNU radio to network-layer performance. If such delays are taken into account, the choice of hardware and software for the underlying physical and data link layers can significantly affect the network-layer performance. This is particularly significant in multi-hop communication as connecting each USRP SU node to a different computer and running separate software code in each computer incur hardware and software delays at each SU intermediate node. Our system architecture uses a single computer running a single software code to coordinate the USRPs. As most processes are performed by a single computer, there are no hardware and software processing delays at each SU intermediate node. Furthermore, a switch provides a seamless control message transmission which is out-of-bound in nature, so that the control message transmission is not affected by data transmission. This is important as each USRP module is only equipped with a single transceiver, and so it cannot transmit data and control messages simultaneously.

III. SYSTEM ARCHITECTURE

This section presents the architecture of the USRP/GNU radio platform for CR networks. Fig. 1 shows an architecture with six USRP SU nodes; while another architecture with ten USRP SU nodes is shown in Fig. 11. In Fig. 1, the SUs are represented as: a source node (m_1), intermediate nodes (m_2, m_3, m_4, m_5), and a destination node (m_6). In our experiment, we show and compare the performance of several route selection schemes in selecting the best possible route out of a number of routes. In the literature, investigations have been conducted with two possible single-hop routes (between the source node and the destination node) in a network with three USRP SU nodes [14], as well as three [13] and four [12] possible routes with a maximum of three hops in a network with six USRP SU nodes. In our work, investigations are conducted with four possible routes with a maximum of

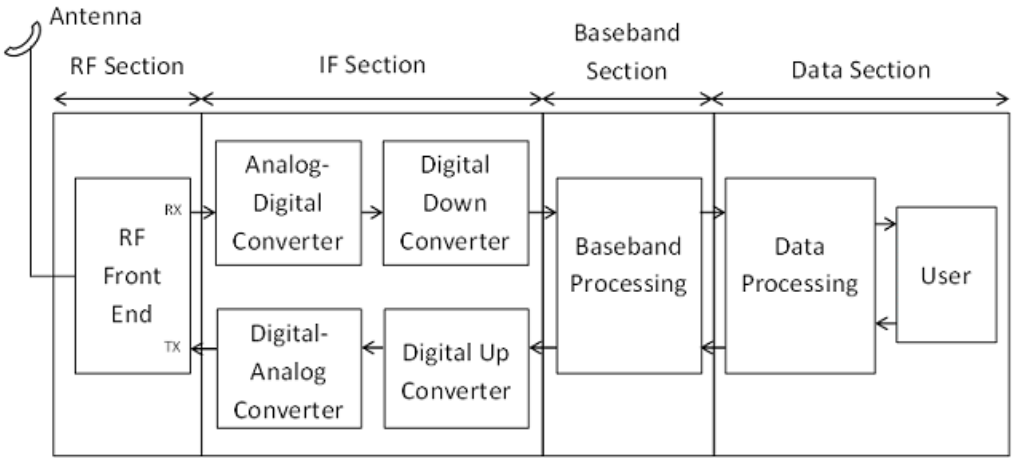

FIGURE 2. Transmit and receive paths in a USRP unit.

three hops in a network with six USRP SU nodes in the 6-node topology (see Fig. 1) and five possible routes with a maximum of four hops in a network with ten USRP SU nodes (see Fig. 11). The PUs are emulated using a Python script, and their interference is represented by dash-line circles in Fig. 1 and Fig. 11. For instance, in Fig. 1, there are four PUs that interferes with the intermediate nodes (i.e., m_2, m_3, m_4, m_5), and each PU can interfere with three SU links from each of the SUs. Specifically, PU interference at intermediate node m_2 affects links $m_1 - m_2$, $m_2 - m_4$ and $m_2 - m_5$, PU interference at intermediate node m_4 affects links $m_2 - m_4$, $m_3 - m_4$ and $m_4 - m_6$, and so on. The USRP hardware provides the RF frontend. GNU radio is a software interface written in Python, and it is installed in the computer. It serves as a signal processing block that perform tasks such as generating and reconfiguring waveforms. In this implementation, it serves two main purposes. Firstly, it houses the decision making engine, and defines the operating environment (e.g., the PUs' activities, which are exponential ON/OFF processes). Secondly, it provides a software interface to the USRP platform. For instance, it receives the channel state information (e.g., channel number and channel availability) from the operating environment. The decision making engine is one of the main components of GNU radio software, and it obtains decision making factors from the operating environment (e.g., PUs' activities), analyzes the decision making factors, and makes selection of the optimal action (e.g., route selection). Both decision making factors and actions are stored in the knowledge base. More details about the decision making engine for RL-based and SL-based approaches can be found in Sections IV-C and IV-D, respectively. The computer and USRP SU nodes are connected to a gigabit Ethernet switch via gigabit wired connections so that the need to connect each USRP SU node to a different computer is not necessary. The gigabit wired connections emulate a common control channel (CCC) used by the USRP SU nodes to exchange route selection messages such as RREQ and RREP. The USRP SU nodes are connected via wireless medium to form a multi-hop CR network. In Fig. 1, the source node m_1 chooses a route that has higher channel available time to the destination node m_6. Further details of the USRP and GNU Radio are presented in the rest of this section.

A. USRP UNIT

Fig. 2 shows the transmit and receive paths in a USRP unit. There are four main sections. Firstly, the radio frequency (RF) section comprises a set of VERT900 antennas, namely RF1 and RF2, which allows transmission and reception in two different channels, respectively. Each antenna is connected to a WBX transceiver daughterboard. The antenna and transceiver daughterboard can transmit and receive radio signals ranging from 824 MHz to 960 MHz. Secondly, the immediate frequency (IF) section consists of analog/digital converters, as well as digital up/down converters. Thirdly, the baseband section performs our proposed route selection schemes. Fourthly, the data section provides a user interface for developing intelligent and knowledge-based mechanisms on the USRP units. This enables a SU network to make the right decisions on route selection in order to enhance network performance.

B. GNU RADIO

Using GNU radio, the functionalities of the transmitters and receivers are represented as flow graphs. Generally speaking, a flow graph starts with a source (e.g., a user datagram protocol (UDP) source) and ends with a sink (e.g., a USRP sink). The schematic representations of the flow graphs for the source, intermediate, and destination nodes are shown in Fig. 3(a), 3(b) and 3(c), respectively. In Fig. 3(a), the source node initiates packet transmission in which a UDP source receives video frames with a payload size of 12 KB from a computer with an IP address 127.0.0.1 via port 1234. The 'Null Pkt is EOF' is set to 'True' to indicate that the end of file occurs when no packet is received. The UDP

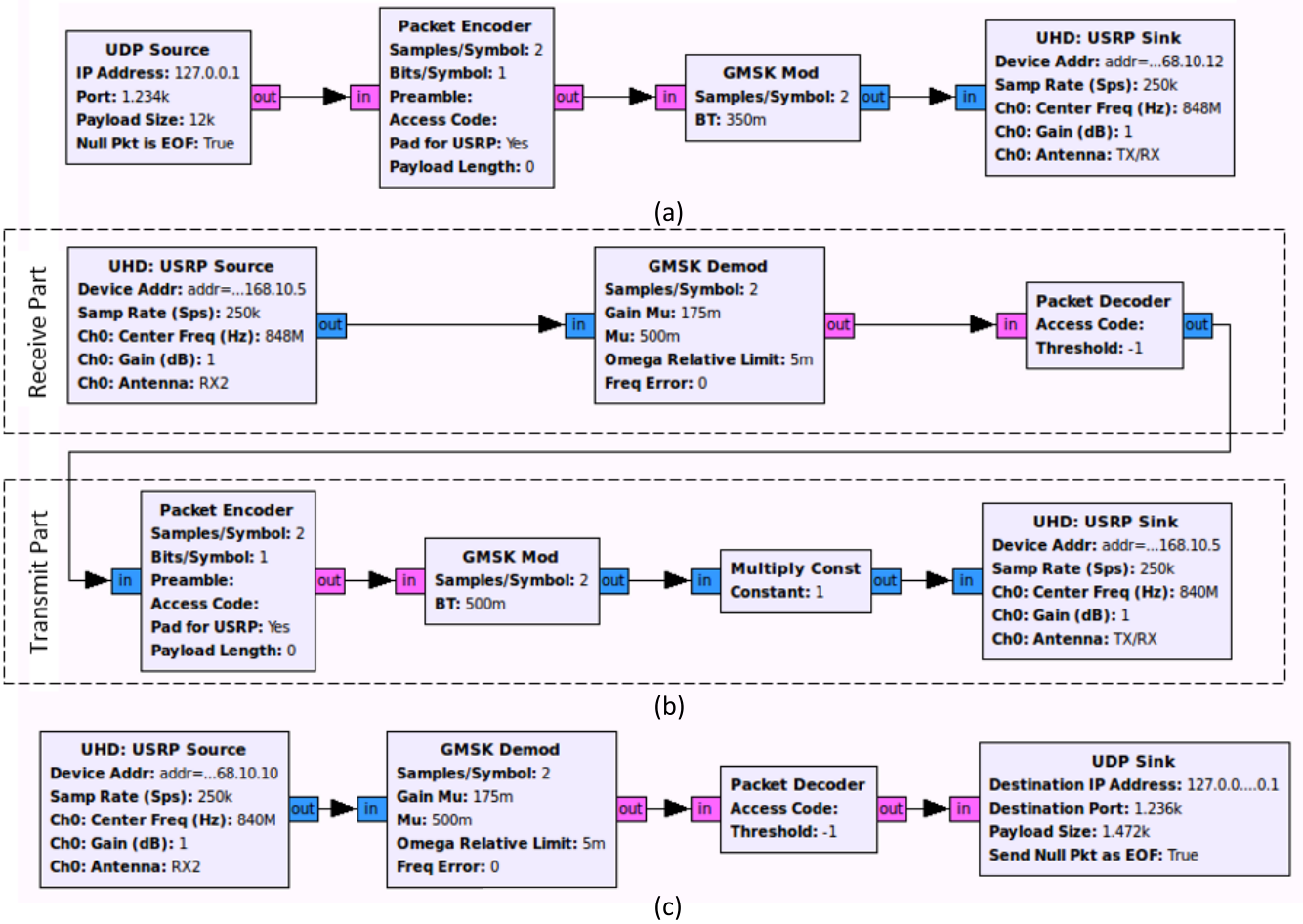

FIGURE 3. Flow graphs for GNU radios. (a) Source node. (b) Intermediate node. (c) Destination node.

source sends the received frames of information, such as video frames, to the packet encoder and modulation blocks. The packet encoder block converts the frames into packets by adding headers, access codes, preamble codes and so on. The 'Samples/Symbol' and 'Bit/Symbol' are set to low values to avoid an error called *underrun* which occurs when the computer is not fast enough to send video frames to the USRPs. The 'Preamble' and Access Code' values are left empty so that preliminarily data is not needed and access code is not assigned to encoded packets. The GMSK modulation converts the packets into signals. Similarly, 'Samples/Symbol' is set to a low value to avoid underrun. Subsequently, the signals are ready for transmission through the USRP sink block via a TX/RX antenna. Specifically, the USRP SU device, which has an IP address 192.168.10.12, sends the signals at a sampling rate of 250,000 samples/seconds using a channel frequency of 848 MHz with a channel gain of 1dB. In Fig. 3(b), the intermediate node receives and retransmits the signals towards the destination node. The intermediate node has two types of paths, namely receive path and transmit path. In the receive path, the USRP SU node, which has an IP address 192.168.10.5, receives the signals at a sampling rate of 250,000 samples/seconds via its antenna RX2 using a channel frequency of 848 MHz with a channel gain of 1dB. The signals are then sent to demodulation and packet decoder blocks which convert the signals into packets, and then back into frames. Similarly, 'Samples/Symbol' is set to a low value in GMSK Demod to avoid an error called underrun, and 'Access Code' value is left empty in Packet Decoder so that access code is not assigned to encoded packets. Next, in the transmit path, packet encoder and modulation blocks reconvert the frames into packets, and modulate them into signals again. Then, the multiply const block amplifies the signals. 'Constant' indicates that the signal power is amplified with the number of times indicated by the value. With a value of 1, there is no amplification as the USRP SU nodes are placed close to each other (see Section V.A). Finally, similar to the USRP sink block in the source node, the USRP sink block in the intermediate node transmits the signals. In Fig. 3(c), the destination node serves as the sink node. The USRP source block receives the signals and sends them to the demodulation and packet decoder blocks which convert the signals into packets and then back into frames. The UDP sink block captures these frames so that applications, such as

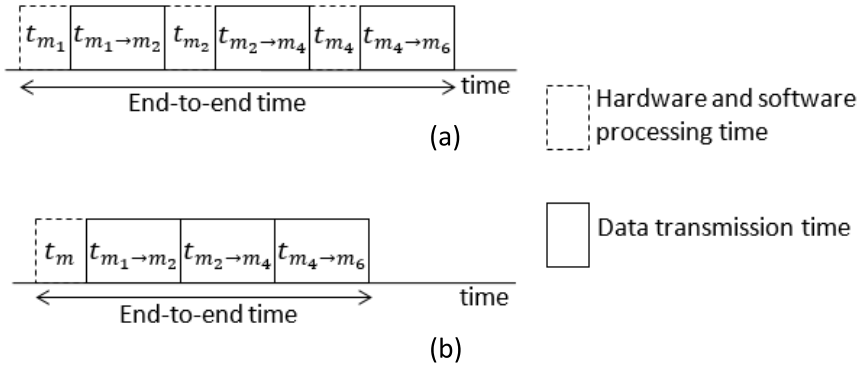

FIGURE 4. Processing time along a route from a source node to a destination node in a multi-hop CR network. (a) Non-switch-based approach. (b) Switch-based approach.

a media player and a web browser, receive the information at the destination node.

IV. ROUTE SELECTION

While routing in multi-hop CR networks has been investigated intensively using simulation platforms (e.g., Qualnet and NS2) [15], there is only perfunctory effort made to implement them on experimental platform. To conduct experiment on route selection in a multi-hop CR network, we propose three schemes. The first two schemes are based on Q-routing [33], which is a popular approach in RL. Although, it has been widely applied in wireless networks and tested through simulation platforms [33], [34], there is lack of experimental investigation. The third scheme is based on the spectrum leasing concept. One of the major issues of the USRP/GNU radio platform is the effects of the underlying delay on network performance. The underlying delay is caused by the processing delays of hardware (i.e., reconfigurations and processes of USRPs and the computer) and software (i.e., initialization and processes of GNU radio or Python codes), and it increases with the number of route breakages [15], [35], [36]. This means that higher number of route breakages increases the hardware and software processing time causing a decline in throughput and packet delivery ratio. In [36], the underlying delay is reported as being in the range of 28.9 ms to 36.9 ms. Since the primary focus of this work is on the network layer, our USRP/GNU radio platform does not consider the underlying processing delays. Section IV-A fur provides further description about delay in USRP/GNU radio. Section IV-B presents our system model. Sections IV-C and IV-D present the route selection schemes based on RL and SL, Section IV-B presents our system model. Sections IV-C and IV-D present the route selection schemes based on RL and SL, respectively.

A. AN OVERVIEW OF THE UNDERLYING LATENCY IN USRP/GNU RADIO PLATFORM

Generally speaking, route selection schemes can be implemented on a USRP/GNU radio testbed using a non-switch-based approach or a switch-based approach. In the non-switch-based approach, each USRP SU node is connected to an individual computer, hence each node incurs the underlying hardware and software delays, which makes it challenging to investigate the network performance achieved by upper layers. In the switch-based approach, which is used in this work, USRP SU nodes are connected to a single gigabit Ethernet switch, which is connected to a single computer. Hence, all the nodes along a route use a single Python code in the switch-based approach, instead of their own individual codes in the non-switch-based approach. This helps the switch-based approach to exclude the underlying hardware and software delays found in the non-switch-based approach. Fig. 4 shows the difference in the end-to-end time for the two approaches whenever a new route is established. Suppose, the source node m_1 selects a new route $m_1 - m_2 - m_4 - m_6$ to the destination node m_6 in Fig. 1. Fig. 4(a) shows the non-switch-based approach in which the hardware and software processing time t_{m_1} is incurred at the source node m_1 for reconfiguration, the data transmission time $t_{m_1 \to m_2}$ is incurred for data transmission from the source node m_1 to the intermediate node m_2, and so on. Fig. 4(b) shows the switch-based approach in which the hardware and software processing time t_m is only incurred at the beginning of a route to reconfigure nodes m_1, m_2, m_4 and m_6. Note that, without any changes of route, the hardware and software processing time is not incurred in both non-switch-based and switch-based approaches as reconfiguration is not required. Nevertheless, a route change is necessary due to the reappearance of PUs' activities.

B. SYSTEM MODEL

The system model consists a set of PUs $P = \{1, 2, \ldots, |P|$ and a set of available channels $C = \{c_1, c_2, \ldots, c_c, \ldots, c_{|C|}\}$, where $|P|$ and $c_{|C|}$ represent the number of PUs and channels, respectively. Fig. 5 shows a network topology, in which the USR P SU nodes are represented by $M = \{m_1, m_2, \ldots, m_N\}$. In the network, there is a single USRP SU source node $m_1 \in M$, a set of intermediate nodes $X_{h,j_h} = m_2, \ldots, m_{N-1} \subseteq M$, and a single destination node $m_N \in M$. A set of routes $K = \{k_1, k_2, \ldots, k_k, \ldots, k_{|K|}$ can be

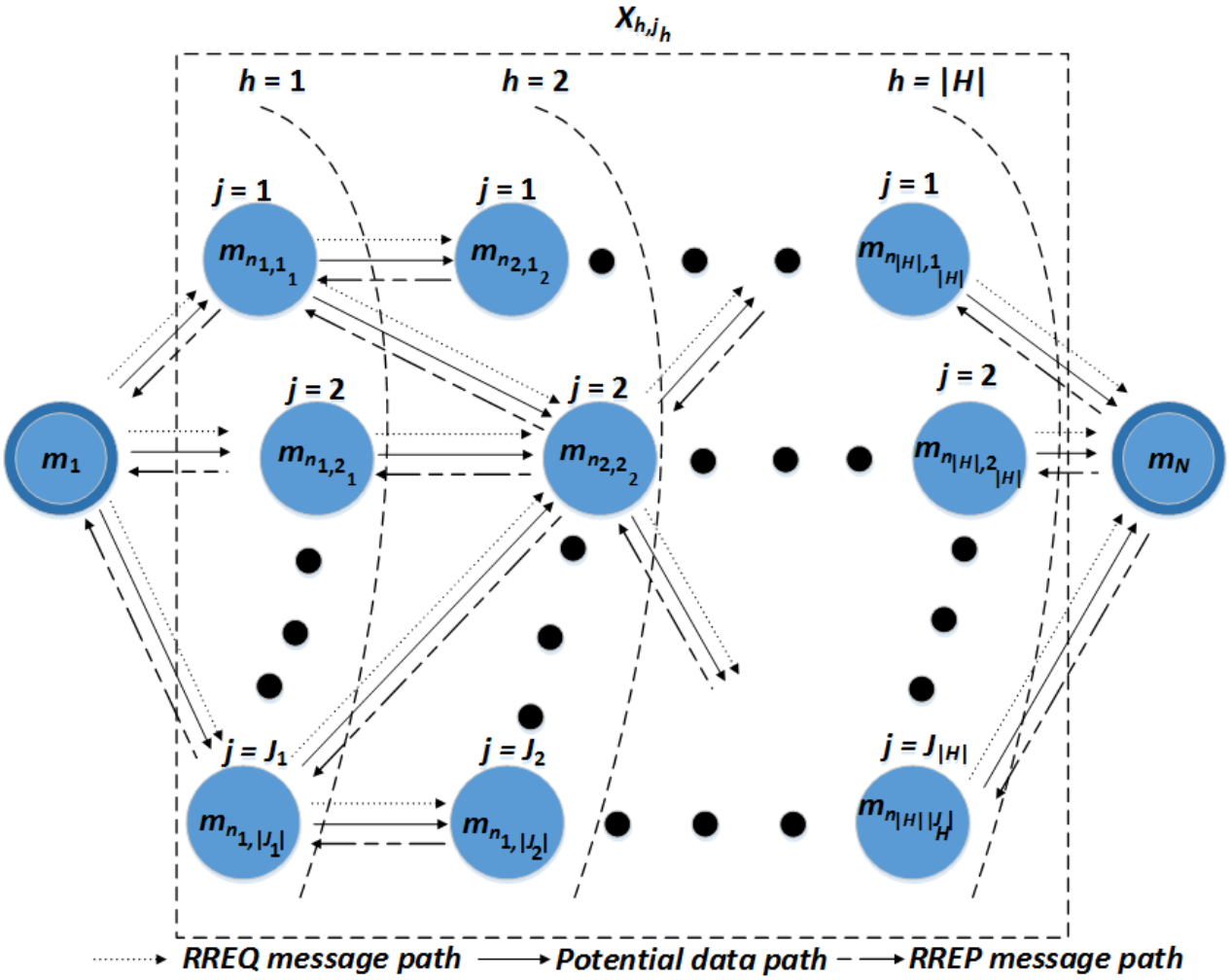

FIGURE 5. A network topology.

established in the network. Each route $k_k \in K$ has a set of links L_k from the source node m_1 to the destination node m_N (e.g., links $m_1 - m_2$, $m_2 - m_4$ and $m_4 - m_6$ in Fig. 1). In the set of intermediate nodes X_{h,j_h}, an intermediate node $m_{n_{h,j_h}}$ has an identification (ID) $n_{h,j_h \in J_h}$, where $J_h = \{1, 2, \ldots, |J_h|\}$ is a set of nodes which are h hops from the source node m_1. Specifically, node $m_{n_{h,j_h}}$ is located at $h \in \{1, 2, \ldots, |H|\}$ hops from the source node m_1, and it is the j_hth node in the set of J_h. The node identification n_{h,j_h} is computed as follows:

$$n_{h,j_h} = h + j_h + \begin{cases} 0 & \text{if } h = 1 \\ \sum_{h=1}^{h-1} (|J_h| - 1) & \text{if } h > 1 \end{cases} \quad (1)$$

The channel selection is dependent on the channel state, which is a two-tuple information comprised of the PU idle/busy state and the channel available time. The PU activity in each channel is either ON (i.e., busy or PUs' activities appear in the channel) or OFF (i.e., idle or no PUs' activity in the channel) state. The ON duration $\tau^p_{c_c,ON}$ and OFF duration $\tau^p_{c_c,OFF}$ of a PU $p \in P$ in its channel $c_c \in C$ follows a Poisson model, and they are exponentially distributed with rates $\lambda^p_{c_c,ON}$ and $\lambda^p_{c_c,OFF}$, respectively. The terms ON duration and PU-ON time, as well as OFF duration and PU-OFF time, are used interchangeably. In this work, we adopt three assumptions on channel selection and access. Firstly, the underlying channel sensing mechanism of the SUs in the data link layer can sense the channel accurately within a channel sensing time window t_s [37] used by PUs to estimate the channel available time of each PUs' channel in longer term. Note that, the time horizon is segregated into time windows, each of which is segregated into channel sensing time window t_s and data transmission time window t_d. Secondly, the neighboring links of SUs use distinct channels in order to avoid data link-layer interference among the respective SUs [38]. Thirdly, as the effects of typical phenomena like fading and shadowing have been well investigated in the literature [39], [40], our focus is on the main characteristic of CR, which is the dynamicity of PUs' activities, so the USRP SU nodes can be placed close to each other as shown in Fig. 10 while emulating the CR environment. These assumptions are

adopted to simplify the underlying physical and data link layers as the focus of our work is on the network layer and the main characteristics CR, specifically the dynamicity of the PUs' activities. A cross-layer approach for physical, data link and network layers are left as our future work.

In our system model, we use route request (RREQ) and route reply (RREP) messages, which have been used in traditional routing schemes (e.g., AODV), to broadcast and gather route(s) information. Suppose, a source node m_1 does not have a route information to its destination node m_N in its routing table. It broadcasts route record list (i.e., $\mathbb{R}_{m_1,m_N} = \emptyset$) using a route request RREQ message in the network to discover all possible routes in the network in two main steps. Firstly, the source node m_1 appends its node ID to the route record list $\mathbb{R}_{m_1,m_N} \leftarrow m_1$, which is included in the RREQ message, and broadcasts it to its next-hop neighbor nodes $m_{n_{1,J_1}}$, which are located in the first hop from the source node m_1, using a CCC. Secondly, each neighbor node in the first hop $m_{n_{1,j_1 \in J_1}}$ appends its node ID to the route record list $\mathbb{R}_{m_1,m_N} \leftarrow (m_1) \cup m_{n_{1,j_1}}$, and broadcasts the respective RREQ message to its next-hop neighbor nodes $m_{n_{2,J_2}}$, which are located in the second hop from the source node m_1. The similar RREQ broadcast mechanism is repeated for each next-hop neighbor node in the remaining hops to form a route k_k of $m_1 - m_{n_{1,j_1}} - \cdots - m_N$. Upon receiving a number of RREQ messages from different possible routes K, the destination node m_N generates a route reply RREP message for each route $k_k \in K$ and sends it back towards the source node m_1 via intermediate nodes X_{h,j_h}. The RREP includes a bottleneck link record $\Gamma^{k_k}_{\beta,t}$, which is the average channel available time at the bottleneck link of a route k_k at time t. The SU node i estimates the average channel available time $\varphi^{i,j,k_k}_{t,c_c,OFF}$ of channel $c_c \in C$ on the link of a route k_k connecting itself and its SU neighbor node j, using (2) [41]:

$$\varphi^{i,j,k_k}_{t,c_c,OFF} = \frac{\lambda^p_{c_c,ON}}{\lambda^p_{c_c,ON} + \lambda^p_{c_c,OFF}} + \frac{\lambda^p_{c_c,OFF}}{\lambda^p_{c_c,ON} + \lambda^p_{c_c,OFF}} \times e^{-(\lambda^p_{c_c,ON} + \lambda^p_{c_c,OFF})^t} \quad (2)$$

Generally speaking, a node i updates the bottleneck link record $\Gamma^{k_k}_{\beta,t}$ if its link to its upstream node j is lower (or $\varphi^{i,j,k_k}_{t,c_c,OFF} < \Gamma^{k_k}_{\beta,t}$). There are two main steps involved in sending the bottleneck link record using the RREP message from a destination node to a source node. Consider a single route $k_k \in K$. Firstly, the destination node m_N initializes the bottleneck link record $\Gamma^{k_k}_{\beta,t}$ with the average channel available time of its link connecting to its upstream neighbor node $m_{n_{|H|,j_{|H|}}} \in X_{h,j_h}$, specifically $\Gamma^{k_k}_{\beta,t} \leftarrow \varphi^{m_N, m_{n_{|H|,j_{|H|}}}, k_k}_{t,c_c,OFF}$. The bottleneck link capacity $\Gamma^{k_k}_{\beta,t}$ is included in the RREP message, and it is sent to its upstream neighbor nodes $m_{n_{|H|,j_{|H|}}} \in X_{h,j_h}$ using a CCC. Secondly, the upstream node $m_{n_{|H|,j_{|H|}}}$ updates the bottleneck link record $\Gamma^{k_k}_{\beta,t}$ in the RREP message if the average channel available time of the link connecting to its upstream neighbor node is lower, specifically $\varphi^{m_{n_{|H|,j_{|H|}}}, m_{n_{|H|-1,j_{|H|-1}}}, k_k}_{t,c_c,OFF} < \Gamma^{k_k}_{\beta,t}$. The remaining nodes in a route k_k follow the same procedure until the RREP message has reached the source node m_1.

C. DECISION MAKING ENGINE FOR RL-BASED SCHEMES

This section proposes two RL-based schemes, namely the traditional RL scheme (or TRL henceforth) and a RL scheme with average Q-value (or ARL henceforth), as the decision-making engine for route selection. In general, TRL and ARL share a similar algorithm except the way in which the Q-values are updated: TRL calculates the Q-values using the traditional approach [42], whereas ARL uses an average Q-value. In general, Q-values constitute knowledge that represents the suitability of an action in a particular state (or operating environment). The decision making engine for the RL-based schemes is shown in Fig. 6, and it is embedded in a SU source node so that it can select a route in which the average channel available time is the highest possible (or the PUs' activities are minimal) in order to increase throughput and packet delivery ratio, as well as to reduce the number of route breakages. The RL-based scheme uses a distributed model (see Section II), in which PUs do not share spectrum occupancy map with SUs, and so the SUs must sense for available channels and calculate the average channel available time.

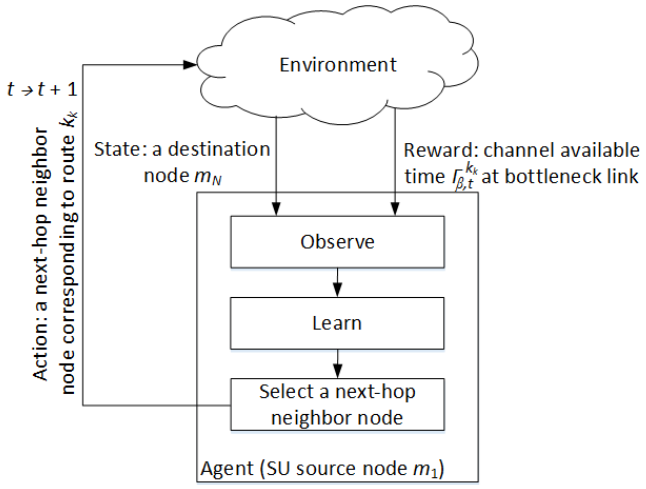

FIGURE 6. Decision making engine for RL-based schemes.

The RL agent receives an update of channel state information of the links of each route in the network (i.e., the average channel available time) using RREQ and RREP (see Section IV-B). Then, the SU source node m_1 (or the RL agent) selects a route based on Algorithm 1. There are three representations, namely state, action and reward. Generally speaking, the SU source node (agent) selects a neighbor node corresponding to a route k_k (action) to its destination node m_N (state) based on the channel state information (reward). Hence, with m_1 being the source node, the state $s^{m_N}_t \in S$ represents a SU destination node, the action $a^{m_{n_{1,j_1}}}_t \in A = \{m_{n_{1,1}}, \ldots, m_{n_{1,|J_1|}}\}$ represents the selection of a neighbor node $m_{n_{1,j_1}}$ of the source node m_1, and the reward R represents

Algorithm 1 Route Selection Mechanism With Q-Value Computation for RL-Based Schemes at SU Node i

/* **Step 1:** */
/* RREQ messages propagation */
if receive $\mathbb{R}_{m_1,m_N}$ and $i! = m_N$ then
  $\mathbb{R}_{m_1,m_N} \leftarrow (\mathbb{R}_{m_1,m_N}) \cup i$
/* RREP message propagation */
else if receive $\mathbb{R}_{m_1,m_N}$ and $i == m_N$ then
  Receive RREP for route $k_k \in K$
  for link (i, j) in k_k do /* node j is an upstream node of node i */
    Estimate $\varphi^{i,j,k_k}_{t,c_c,OFF}$ using (2)
    /* mechanism in a destination node */
    if $i == m_N$
      $\Gamma^{k_k}_{\beta,t} \leftarrow \varphi^{i,j,k_k}_{t,c_c,OFF}$
    /* mechanism in an intermediate nodes and source node */
    else if $(i! = m_1 || i! = m_N)$
      if $\varphi^{i,j,k_k}_{t,c_c,OFF} >= \Gamma^{k_k}_{\beta,t}$
        $\Gamma^{k_k}_{\beta,t} \leftarrow \Gamma^{k_k}_{\beta,t-1}$ /* $\Gamma^{k_k}_{\beta,t}$ is not updated */
      else if $\varphi^{i,j,k_k}_{t,c_c,OFF} < \Gamma^{k_k}_{\beta,t}$
        $\Gamma^{k_k}_{\beta,t} \leftarrow \varphi^{i,j,k_k}_{t,c_c,OFF}$ /* $\Gamma^{k_k}_{\beta,t}$ is updated */
      end if
    end if
    if $i! = m_1$
      Send RREP with $\Gamma^{k_k}_{\beta,t}$ to upstream node i
    end if
  end for
end if
/* **Step 2:** Source node m_1 updates Q-value */
Update Q-value $Q^{m_1}_{t+1}(s^{m_N}_t, a^{m_{n_{1,j_1}}}_t)$ using
  (3) forTRL
  (5) forARL
/* **Step 3:** Source node m_1 determines action */
Determine a^*_t using (4)

TABLE 2. RL-based model embedded in the SU source node m_1.

State	$s^{m_N}_t \in S = \{m_1, m_{n_{1,1_1}}, m_{n_{1,2_1}}, \ldots, m_N\}$, where each state $s^{m_N}_t$ represents a potential SU destination node
Action	$a^{m_{n_{1,j_1}}}_t \in A = \{m_{n_{1,1_1}}, m_{n_{1,2_1}}, \ldots, m_{n_{1,\lvert J_1\rvert}} \lvert m_{n_{1,1_1}} \in k_k\}$, where A is the set of neighboring nodes of source node m_1, and the action $a^{m_{n_{1,j_1}}}_t$ represents the selection of a node $m_{n_{1,j_1}}$, and hence the selection of route $k_k \in K$
Reward	Channel available time $\Gamma^{k_k}_{\beta,t+1}$ at the bottleneck link for a selected route k_k at time $t+1$ between a source node and a destination node

the positive or negative consequence of the action taken in the state, which is the highest channel available time $\Gamma^{k_k}_{\beta,t}$ at the bottleneck link of a route k_k connecting a source node and a destination node, and so the reward varies with the dynamicity of the PUs' activities. Table 2 shows the RL-based model embedded in the SU source node m_1.

Algorithm 1 shows three steps to select a route in the proposed RL schemes. In Step 1, the RL agent interacts with the operating environment using RREQ and RREP messages to obtain updated information about a set of routes K, including channel state information of the routes in the network (i.e., the average channel available time of each link along a route between a source node m_1 and a destination node m_N). In Step 2, based on the average channel available time, the RL agent computes the Q-value of each route $k_k \in K$ in the network. In Step 3, the RL agent selects a route k_k that offers the highest Q-value. Subsequently, the RL agent uses the selected route k_k for data transmission, and the route consists of multiple links operating on different channels. When the PUs' activities reappear in any of these channels assigned to one of the links along the route k_k, the route is considered broken. This is followed by the transmitting node (or upstream node) of the respective link sending a route breakage message to the source node. The source node selects another route in the next time window $t+1$.

1) TRADITIONAL RL-BASED SCHEME

In TRL, the SU source node selects a next-hop neighbor node $a^{m_{n_{1,j_1}}}_t$, which corresponds to a route k_k, leading towards its destination node $s^{m_N}_t$ at time t. It receives its reward in the form of channel available time $\Gamma^{k_k}_{\beta,t+1}$ at the bottleneck link, and updates the corresponding Q-value $Q^{m_1}_t(s^{m_N}_t, a^{m_{n_{1,j_1}}}_t)$ for the state-action pair at time $t+1$ as follows:

$$Q^{m_1}_{t+1}\left(s^{m_N}_t, a^{m_{n_{1,j_1}}}_t\right) \leftarrow (1-\alpha) \times Q^{m_1}_t\left(s^{m_N}_t, a^{m_{n_{1,j_1}}}_t\right) + \alpha \times \Gamma^{k_k}_{\beta,t+1} \quad (3)$$

where $0 \leq \alpha \leq 1$ is the learning rate. When α is higher, the Q-value is more dependent on the current knowledge (or the reward, which is the channel available time $\Gamma^{k_k}_{\beta,t+1}$ at the bottleneck link of route k_k at time $t+1$); and when α is lower, the Q-value is more dependent on the previous knowledge (or the Q-value $Q^{m_1}_t(s^{m_N}_t, a^{m_{n_{1,j_1}}}_t)$ at time t). Based on (3), the source node m_1 selects the next-hop neighbor node a^*_t, which corresponds to a route k_k, with the highest Q-value as follow:

$$a^*_t = argmax_{a \in A} Q^{m_1}_t(s^{m_N}_t, a) \quad (4)$$

2) RL-BASED SCHEME WITH AVERAGE Q-VALUE

The average Q-value based route selection scheme (ARL) has been shown to improve stability in simulation setting [43], [44]. In this approach, the average Q-value $\bar{Q}^{m_1}_t(s^{m_N}_t, a^{m_{n_{1,j_1}}}_t)$ is calculated, and it is used in Q-function $Q^{m_1}_{t+1}\left(s^{m_N}_t, a^{m_{n_{1,j_1}}}_t\right)$ to select more stable routes as follows [44]:

$$Q^{m_1}_{t+1}\left(s^{m_N}_t, a^{m_{n_{1,j_1}}}_t\right) \leftarrow (1-\alpha) \times Q^{m_1}_t\left(s^{m_N}_t, a^{m_{n_{1,j_1}}}_t\right) + \alpha \times (\Gamma^{k_k}_{\beta,t+1} + \bar{Q}^{m_1}_t(s^{m_N}_t, a^{m_{n_{1,j_1}}}_t)) \quad (5)$$

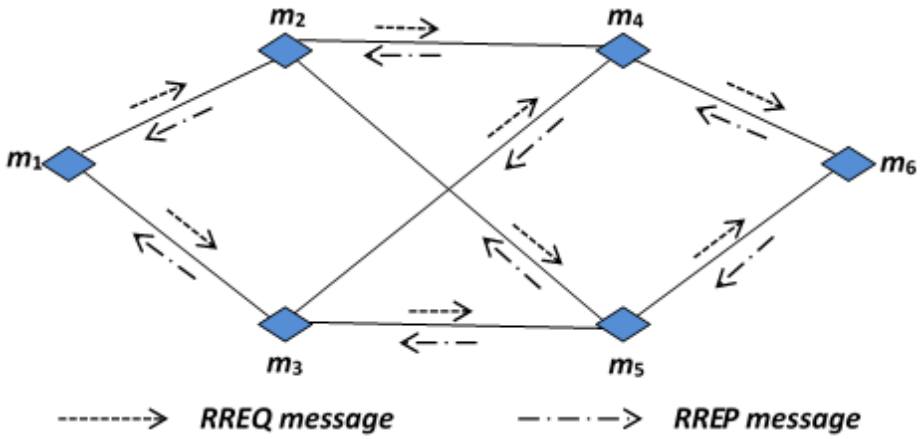

FIGURE 7. A 6-node topology for experimental study with RREQ and RREP message exchanges.

The average Q-value $\bar{Q}_t^{m_1}(s_t^{m_N}, a_t^{m_{n_1,j_1}})$ is a ratio of the sum of all Q-values $Q_{t+1}^{m_1}\left(s_t^{m_N}, a_t^{m_{n_1,j_1}}\right)$ up to time t to the total number of times if the route k_k is selected for transmission (similarly for the case if the route k_k is not selected), and it is calculated as follows:

$$\bar{Q}_t^{m_1}\left(s_t^{m_N}, a_t^{m_{n_1,j_1}}\right) = \begin{cases} \dfrac{\sum_{k_k \epsilon K} P(k^+ = 1|k_k) Q_t^{m_1}(s_t^{m_N}, a_t^{m_{n_1,j_1}})}{\sum_{k_k \epsilon K} P(k^+ = 1|k_k)}; \\ \quad k_k \text{ is selected} \\ \dfrac{\sum_{k_k \epsilon K} P(k^- = 0|k_k) Q_t^{m_1}(s_t^{m_N}, a_t^{m_{n_1,j_1}})}{\sum_{k_k \epsilon K} P(k^- = 0|k_k)}; \\ \quad k_k \text{ is not selected} \end{cases} \tag{6}$$

where $P(k^+ = 1|k_k)$ is the probability that route k_k is selected for transmission; and similarly, $P(k^- = 0|k_k)$ is the probability that route k_k is not selected for transmission. Also, $\sum_{k_k \epsilon K} P(k^+ = 1|k_k)$ represents the sum of all the probability that route k_k is selected for transmission, which determines the total number of times the route is selected; and similarly, $\sum_{k_k \epsilon K} P(k^- = 0|k_k)$ represents the sum of all the probability that route k_k is not selected for transmission, which determines the total number of times the route is not selected. With the consideration of average Q-value, which is dependent on the probabilities of selecting (or not selecting) a route, the RL agent can select a stable route that has been selected for transmission in the past repeatedly. Based on (5), the source node selects the next-hop neighbor node a_t^*, which corresponds to route k_k, with the highest Q-value using (4).

3) AN ILLUSTRATION OF RL-BASED SCHEMES

Consider an experimental setup to examine the RL-based schemes in a multi-hop CR network as shown in Fig. 7, which is a simplified topology representation of Fig. 1. There are six SUs: a source node m_1, intermediate nodes m_2, m_3, m_4, m_5, and a destination node m_6. Initially, the source node m_1 has no route information leading to its destination node m_6 in the route record list (i.e., $\mathbb{R}_{m_1,m_6} = \emptyset$). Then, the source node m_1 appends its own ID to the route record list (i.e., $\mathbb{R}_{m_1,m_6} \leftarrow m_1$) in a newly generated RREQ message, and broadcasts the RREQ message to its neighboring nodes m_2 and m_3 using a CCC to discover routes leading to the destination node m_6. When the neighboring nodes m_2 and m_3 receive separate RREQ messages from the source node m_1, node m_2 appends its ID to the route record list (i.e., $\mathbb{R}_{m_1,m_6} \leftarrow (m_1) \cup m_2$), and node m_3 does the same (i.e., $\mathbb{R}_{m_1,m_6} \leftarrow (m_1) \cup m_3$). Next, nodes m_2 and m_3 forward their respective RREQ messages via CCC to their respective next-hop neighboring nodes m_4 and m_5. The similar procedure is repeated at nodes m_4 and m_5 until the RREQ messages reach destination node m_6. In Fig. 7, the destination node m_6 receives four possible routes, namely route k_1 $(m_1 - m_2 - m_4 - m_6)$, route $k_2(m_1 - m_2 - m_5 - m_6)$, route $k_3(m_1 - m_3 - m_4 - m_6)$ and route $k_4(m_1 - m_3 - m_5 - m_6)$.

Subsequently, the destination node m_6 generates RREP messages which traverse back towards the source node m_1 using the reversed route given in the RREQ messages. Consider route k_1. The destination node m_6 obtains the average channel available time $\varphi_{t,c_c,OFF}^{m_4,m_6,k_1}$ of the link $m_4 - m_6$ using (2) and updates the channel available time of the bottleneck link $\Gamma_{\beta,t}^{k_1}$ in the RREP message. Next, the destination node m_6 sends the RREP message to its upstream node m_4. When node m_4 receives the RREP message, it obtains the average channel available time $\varphi_{t,c_c,OFF}^{m_2,m_4,k_1}$ of the link $m_2 - m_4$. If $\varphi_{t,c_c,OFF}^{m_2,m_4,k_1}$ is smaller than the channel available time of the bottleneck link $\Gamma_{\beta,t}^{k_1}$ in the RREP message (or $\varphi_{t,c_c,OFF}^{m_2,m_4,k_1} < \Gamma_{\beta,t}^{k_1}$), then node m_4 updates the channel available time of the bottleneck link (or $\Gamma_{\beta,t}^{k_1} = \varphi_{t,c_c,OFF}^{m_2,m_4,k_1}$) in the RREP message; otherwise the channel available time of the bottleneck link remains the same (or $\Gamma_{\beta,t}^{k_1} = \varphi_{t,c_c,OFF}^{m_4,m_6,k_1}$). The same process is repeated until the RREP message reaches the source node m_1.

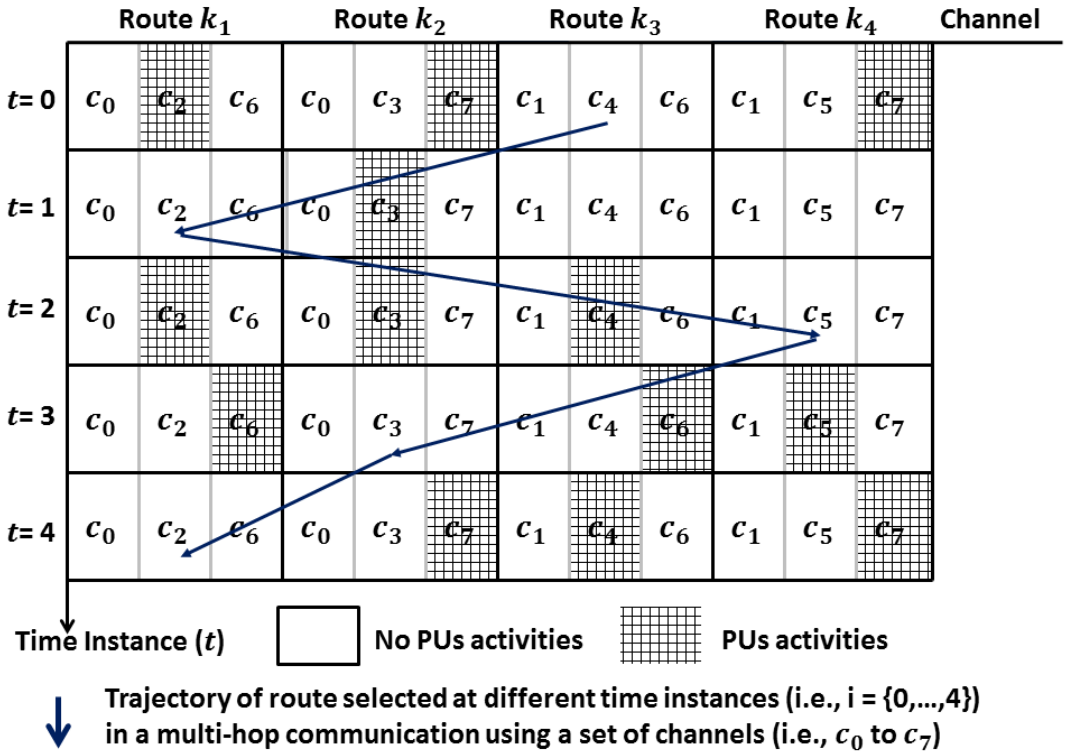

FIGURE 8. Trajectory of route selection at different time instances.

Upon receiving RREP messages for the four routes, the source node m_1 computes the Q-value of each route using (3) for TRL (or (5) for ARL), and selects the route which has the highest Q-value using (4). Fig. 8 shows the trajectory of a selected route on the basis of the highest Q-value at different time instances. For instance, at time instance $t = 0$, route k_3 is selected, as there are no PUs' activities in the respective channels of the links (e.g., channel c_1 is used in link $m_1 - m_3$, channel c_4 in $m_3 - m_4$, and channel c_6 in $m_4 - m_6$) of the route, and so it has the highest Q-value. Whereas, route k_1, route k_2, and route k_4 are not chosen due to the presence of PUs' activities in channels c_2 and c_7, which are both chosen by the links $m_2 - m_4$ and $m_5 - m_6$ of those routes.

D. DECISION MAKING ENGINE FOR SL-BASED SCHEME

The decision making engine for SL-based scheme is shown in Fig. 9, and it is embedded in a SU source node so that it can select the best possible route from a source node to a destination node in order to increase throughput and packet delivery ratio, as well as reduce the number of route breakages. The SL-based scheme uses a centralized model (see Section II), in which the PUs share their spectrum occupancy map (i.e., ON duration $\tau^p_{c_c,ON}$ and OFF duration $\tau^p_{c_c,OFF}$) with SUs located within their transmission range. In practice, the PUs can gain monetary rewards from SUs for sharing the spectrum occupancy map with them. The PUs only allow SUs to use their channels whenever the PUs are in their inactive state (i.e., OFF duration $\tau^p_{c_c,OFF}$). It is also beneficial for

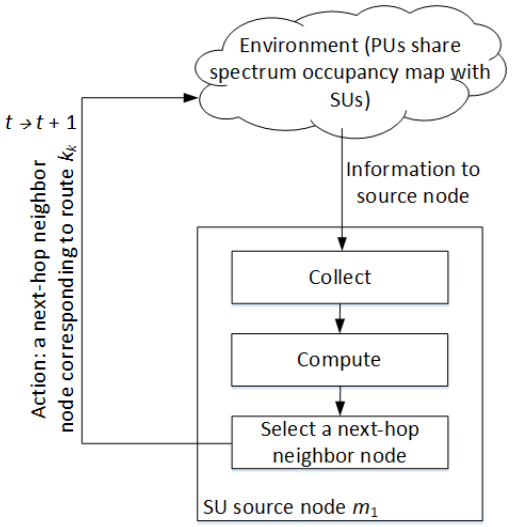

FIGURE 9. Decision making engine for SL-based scheme.

SUs to be aware of the OFF duration $\tau^p_{c_c,OFF}$ time of PUs, which provides them with the exact channel available time (i.e., $\mathfrak{e}^{i,j,k_k}_{t,c_c,OFF} \leftarrow \tau^p_{c_c,OFF}$) for transmission of their packets. For instance, SU node i is in the transmission range of PU p. The PU p shares its $\tau^p_{c_c,OFF}$ time, which is the OFF duration of channel c_c, with SU node i. So, the SU node i can use this exact channel available time $\mathfrak{e}^{i,j,k_k}_{t,c_c,OFF}$ for packet transmission with its neighboring SU node j, which corresponds to route k_k. Algorithm 2 shows three steps to

Algorithm 2 Route Selection Mechanism for the SL-Based Scheme at Node i

```
/* Step 1 */
Receive (℘^{i,j,k_k}_{t,c_c,OFF} ← τ^p_{c_c,OFF}) from PUs
/* Step 2 */
/* Source node m_1 initiates the RREQ messages propagation */
if receive ℝ_{m_1,m_N} and i! = m_N then
    ℝ_{m_1,m_N} ← (ℝ_{m_1,m_N}) ∪ i
/* RREP message propagation */
else if receive ℝ_{m_1,m_N} and i == m_N then
    Receive RREP for route k_k ∈ K
    for link (i, j) in k_k do /* node j is an upstream node of node i */
        /* mechanism in a destination node */
            if i == m_N
                Γ^{k_k}_{β,t} ← ℘^{i,j,k_k}_{t,c_c,OFF}
        /* mechanism in an intermediate node and source node */
        else if (i! = m_1 || i! = m_N))
            if ℘^{i,j,k_k}_{t,c_c,OFF} >= Γ^{k_k}_{β,t}
                Γ^{k_k}_{β,t} ← Γ^{k_k}_{β,t-1}
            else if ℘^{i,j,k_k}_{t,c_c,OFF} < Γ^{k_k}_{β,t}
                Γ^{k_k}_{β,t} ← ℘^{i,j,k_k}_{t,c_c,OFF}
            end if
        end if
        if i! = m_1
            Send RREP with Γ^{k_k}_{β,t} to upstream node i
        end if
    end for
/* Step 3 */
Determine a*_t using (8)
```

select a route in the proposed SL scheme. In Step 1, every SU node (i.e., $m_1 \in M$ or $X_{h,j_h} \in M$ or $m_N \in M$) receives spectrum occupancy maps (i.e., ON duration $\tau^p_{c_c,ON}$ and OFF duration $\tau^p_{c_c,OFF}$) from PUs within their respective transmission ranges. In Step 2, the SU source node m_1 uses RREQ and RREP messages to collect the exact channel available time $\wp^{i,j,k_k}_{t,c_c,OFF}$ of the links along all possible routes K. Based on $\wp^{i,j,k_k}_{t,c_c,OFF}$, the SU source node computes the bottleneck link $\Gamma^{k_k}_{\beta,t}$ of each route $k_k \in K$ in the network. In Step 3, the SU source node selects a route k_k that offers the highest channel available time at its bottleneck link. Subsequently, the source node uses the selected route k_k for data transmission, and the route consists of multiple links operating on different channels. When the PUs' activities reappear in any of these channels assigned to one of the links along the route k_k, the route is considered broken. This is followed by the transmitting node (or upstream node) of the respective link sending a route breakage message to the source node. The source node selects another route in the next time window $t + 1$.

In contrast to the RL-based decision making engine, which is embedded in the SU source node only, the SL-based decision making engine is embedded in each SU node. The PUs provide their spectrum occupancy map (i.e., ON duration $\tau^p_{c_c,ON}$ and OFF duration $\tau^p_{c_c,OFF}$ for their respective channel c_c) to SU nodes that are within their transmission range. The SU source node m_1 receives these updates using RREQ and RREP. Upon receiving the exact channel available time $\wp^{i,j,k_k}_{t,c_c,OFF}$ of every link (e.g., a link between SU node i and SU node j is $m_i - m_j$), the SU source node m_1 obtains the minimum channel available time $\Gamma^{k_k}_{\beta,t}$ at the bottleneck link along the route k_k from source node m_1 to destination node m_N, which can be computed as follows:

$$\Gamma^{k_k}_{\beta,t} = \underset{(i,j)\in k}{\operatorname{argmin}} \wp^{i,j,k_k}_{t,c_c,OFF} \quad \forall\, k_k \in K \tag{7}$$

Based on (7), the source node selects the next-hop neighbor node $a \in m_{n_{1,j_1}}$, which corresponds to route k_k, that offers the highest minimum channel available time $\Gamma^{k_k}_{\beta,t}$ at its bottleneck link as follows:

$$a^*_t = \underset{k_k \in k}{\operatorname{argmax}} \Gamma^{k_k}_{\beta,t} \tag{8}$$

1) AN ILLUSTRATION OF SL-BASED SCHEME

Consider an experimental setup to examine the SL-based scheme in a multi-hop CR network as shown in Fig. 7. The SL-based scheme uses a centralized model, in which the PUs share their respective spectrum occupancy map (i.e., ON duration $\tau^p_{c_c,ON}$ and OFF duration $\tau^p_{c_c,OFF}$) with SUs located within their respective transmission range. So, the source node m_1 needs to collect information about the routes and the exact channel access time $\wp^{i,j,k_k}_{t,c_c,OFF}$ of the links in the network. The SL-based scheme shares similar mechanism with the RL-based scheme. The only exception is that, in the SL-based scheme, the SUs receive the exact channel access time rather than the estimated average channel available time $\varphi^{i,j,k_k}_{t,c_c,OFF}$ (see (2)). Next, upon receiving RREP messages for the four routes, the source node m_1 obtains the minimum channel available time of the bottleneck link of each route, and selects the route with the highest minimum channel available time using (8).

V. EXPERIMENT AND EVALUATION

This section presents experimental setup and performance evaluation. The experimental parameters for both topologies are shown in Table 3.

A. EXPERIMENTAL SETUP

Two experimental scenarios, namely a 6-node topology (see Fig. 1) and a 10-node topology (see Fig. 11), for multi-hop CR networks are considered. The 6-node topology and 10-node topology have six and ten USRP SU nodes, respectively. In this paper, we deploy topologies of up to 10 USRP SU nodes, which extend existing implementations [12], [13] with more nodes. Fig. 10 shows the physical deployment of a

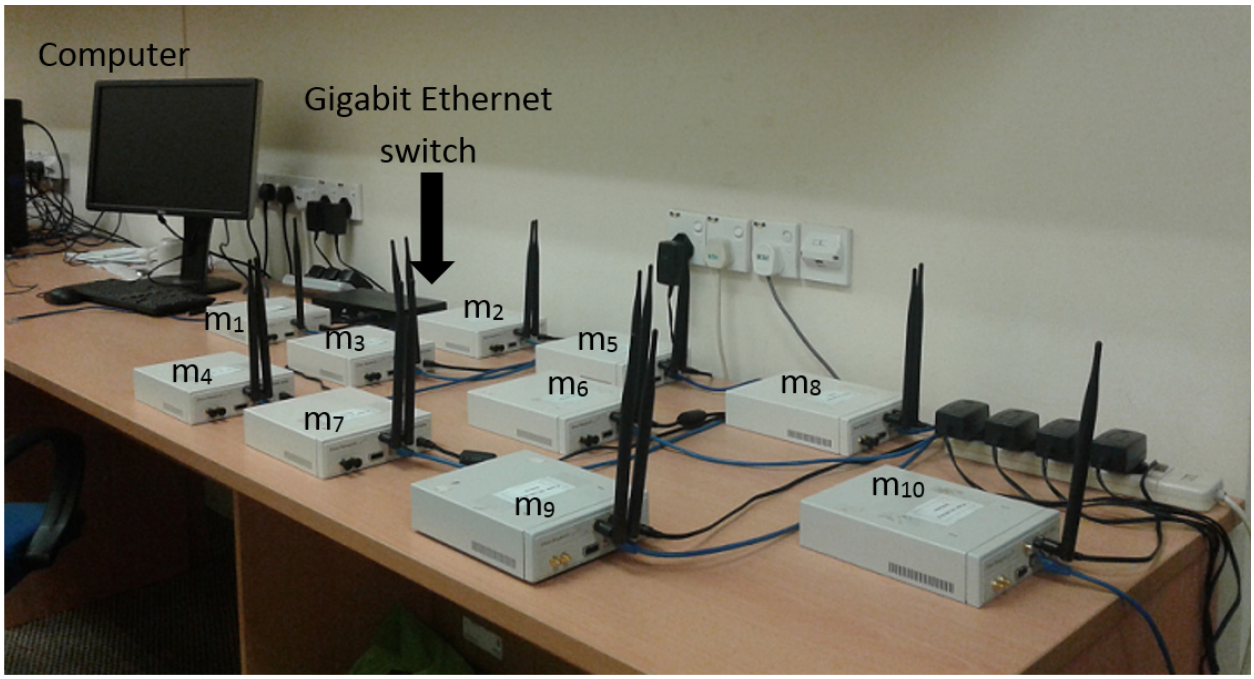

FIGURE 10. Physical deployment of a 10-node topology.

TABLE 3. Experimental parameters.

Category	Adjustable Parameter	Value	
		6-node Topology	10-node Topology
PU	Number of PUs	4	6
	PUs' activities model	Exponential ON-OFF model	
	$\lambda^{p}_{c_c,ON}$	15 s	
	$\lambda^{p}_{c_c,OFF}$	{10, 20, 30, 40, 50, 60, 70, 80} s	
SU	Number of USRP SU nodes	6	10
Antenna	Carrier frequency range	824 MHz – 960 MHz	
Network (USRP)	Number of channels	8	13
	Modulation type	GMSK	
	Supported bandwidth	~40 MHz	
	Throughput	8 Mbps	
	Transport layer	UDP	
	Experiment duration	300 s	
RL	Learning rate α	{0.1, 0.3, 0.5, 0.7, 0.9}	

10-node topology in which the USRP SU nodes are connected via a gigabit Ethernet switch to a computer, which runs the GNU radio software that loads the Python program into the USRP SU nodes (see Section III-A). The USRP/GNU Radio testbed is setup in an indoor environment (i.e., a hall with concrete walls) where the USRP SU nodes are placed on a 2 feet $\times$ 3.8 feet table. In Fig. 10, based on our assumptions (see Section IV.B), the USRP SU nodes are placed close to each other with a maximum distance of 4.5 inch between a pair of USRP SU nodes while emulating the main characteristic (i.e., dynamicity of PUs' activities) of a CR environment. In addition, the neighboring links of SUs use distinct channels in order to avoid data link-layer interference among the respective SUs. In this regard, each transceiver uses two distinct frequencies for transmission and reception, and a guard-band of 8 MHz is used in between the two frequencies in order to avoid interference. The computer runs a media player application (i.e., VLC) in a server and client mode and feeds the video into a USRP SU source node. User Datagram Protocol (UDP) is used so that the effects of congestion window in Transmission Control Protocol (TCP) are not considered. The PUs' activities are emulated using an exponential ON-OFF model (see Section IV-B) within a Python code. Throughout the experiment, the rate of the ON time duration of PU p in each of its channel $c_c \in C$ is a constant $\lambda^{p}_{c_c,ON} = 15$ s, and the rate of the OFF time duration of PU p in each of its channel $c_c \in C$ is a variable $\lambda^{p}_{c_c,OFF}$ ranging from 10 s to 80 s [12], [13]. This means that the channel utilization of PU ranges from 16% (or 15/95) to 60% (or 15/25). The channels with a channel utilization of PU of more than 60% are not considered in our experiment as the SUs can highly interfere with the PUs. Each USRP SU node is equipped with VERT900 antennas [45], and they can operate in frequency ranging from 824 MHz to 960 MHz. As temporal variability occurs in the real-world wireless environment, each experiment is repeated 15 times, and each experiment runs for a duration of 300 s.

The two topologies are selected in order to analyze the QoS performance of the proposed schemes, and to investigate the scalability of the network with 6 and 10 USRP

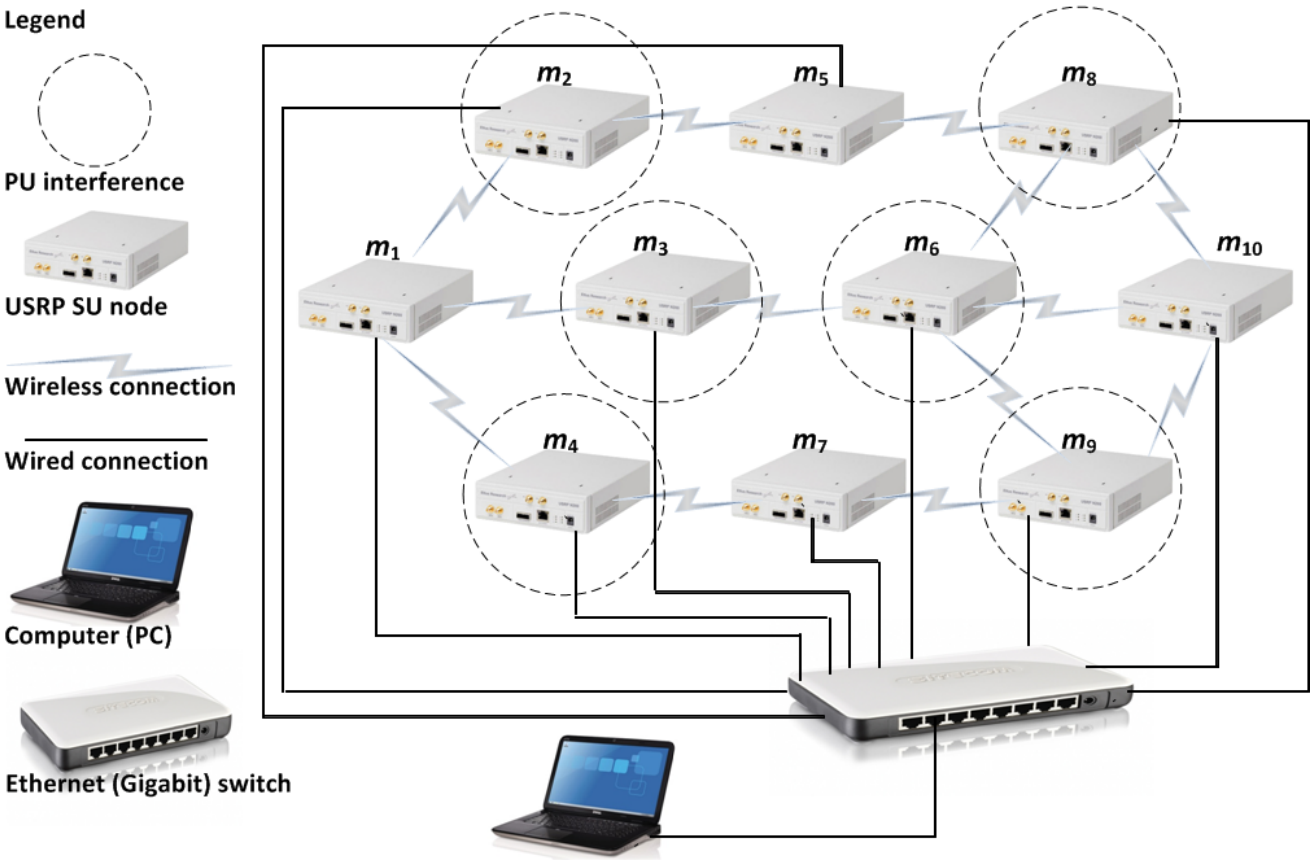

FIGURE 11. A 10-node topology consists of ten USRP SU nodes.

SU nodes. In the 6-node topology as shown in Fig. 1, there are 6 USRP SU nodes (i.e., a source node m_1, intermediate nodes m_2, m_3, m_4, m_5, and a destination node m_6). There are 4 PUs (see Section IV-B for the model of PUs' activities), and each of them interferes with the intermediate nodes (i.e., m_2, m_3, m_4, m_5). Hence, each PU can interfere with 3 SU links from each of the SUs within the PU's coverage area. There are four possible routes from the source node m_1 to the destination node m_6, namely route k_1 $(m_1 - m_2 - m_4 - m_6)$, route $k_2(m_1 - m_2 - m_5 - m_6)$, route $k_3(m_1 - m_3 - m_4 - m_6)$ and route $k_4(m_1 - m_3 - m_5 - m_6)$. In the 10-node topology as shown in Fig. 11, there are 10 USRP SU nodes: a source node m_1, intermediate nodes $m_2, m_3, m_4, m_5, m_6, m_7, m_8, m_9$, and a destination node m_{10}. There are 6 PUs that interfere with the intermediate nodes (i.e., $m_2, m_3, m_4, m_6, m_8, m_9$). At nodes m_2, m_3 and m_4 a PU can interfere with 2 SU links from each of the SUs that lie within the PU's coverage area; while at m_8 and m_9, a PU can interfere with 3 SU links from each of the SUs; and at m_6, a PU can interfere with 4 SU links from each of the SUs. In both topologies, the number of PUs constitutes approximately 60% of the total number of SU nodes. There are five possible routes from the source node m_1 to the destination node m_{10}, namely route k_1 $(m_1 - m_2 - m_5 - m_8 - m_{10})$, route $k_2(m_1 - m_3 - m_6 - m_8 - m_{10})$, route $k_3(m_1 - m_3 - m_6 - m_{10})$, route $k_4(m_1 - m_3 - m_6 - m_9 - m_{10})$ and route $k_5(m_1 - m_4 - m_7 - m_9 - m_{10})$. Since SU-SU interference is not considered in this investigation based on our assumption (see Section IV-B for explanation), neighboring links between a pair of SUs use distinctive channels in order to avoid interference among the respective SUs. We use 8 and 13 channels in the 6-node topology and 10-node topology, respectively (see Section IV-B). The guard-band between two channels is set to 8 MHz for smooth video transmission and to avoid inter-channel interference.

B. PERFORMANCE EVALUATION

This subsection presents performance evaluation including experiment ordinates, performance metrics, complexity analysis and results.

1) EXPERIMENT ORDINATES AND PERFORMANCE METRICS

The experiment ordinate is the PUs' OFF time. The PUs' OFF time $\lambda^{p}_{c_c,OFF}$ is the time duration in a time window during which the PUs are inactive. The performance metrics are packet delivery ratio, the number of route breakages, and throughput. The packet delivery ratio is the total number of packets received by the destination node to the total number of packets sent by the source node. A route breakage happens whenever a PU reappears in the channel of any of the links

in the route for packet transmission from a source node to a destination node. Lastly, the throughput represents the effectiveness of the network in delivering data packets from a source node to a destination node, and it is measured in bits per second (bps).

2) COMPLEXITY ANALYSIS

This section presents the complexity analysis of our proposed RL-based and SL-based schemes in terms of message and time complexities. The message complexity $\mathcal{M}$ is defined as the number of messages exchanged in the network in order to obtain updated information (i.e., channel state information of the routes) about a set of routes K. The channel state information consists of the average channel available time of each link along a route from a source node m_1 to a destination node m_N. When a SU sends a message to each of its neighboring SUs, a single message is incurred, and so the message complexity $\mathcal{M}$ is increased by one. The time complexity T is defined as the number of time steps incurred to perform route selection, which covers finding the number of available routes in the network, selecting a route and switching from a broken route, which may occur due to the re-appearance of the PUs' activities at the bottleneck link of the route to another one. We assume discrete time steps. One time step is the time incurred between the transmission of a message from a SU sender node and the reception of the message at its SU receiver node. Suppose, a SU source node generates a RREQ message and broadcast it towards its neighboring SU nodes. This process continues until the message reaches its SU destination node. Denote the average number of neighbor nodes for each node by η_i, and the intermediate nodes are up to h hops away from the SU source node. So, the whole process of RREQ message propagation takes $\mathcal{M}_{RREQ} = \eta_i \times (h+1)$ messages and $T_{RREQ} = h+1$ time steps. Upon receiving RREQ message, the SU destination node generates RREP message and sends it back on the reverse route $k_k \epsilon K$ that the RREQ message has traversed. We can denote the number of intermediate nodes from a SU source node to a SU destination node along a route k_k involved in RREQ is equal to the number of hops h. So, the whole process of RREP message propagation takes $\mathcal{M}_{RREP} = \sum_{k_k \in K} (h+1)^{k_k}$ messages and $T_{RREP} = h+1$ time steps. So, the total of message complexity in our proposed schemes is $\mathcal{M} = \mathcal{M}_{RREQ} + \mathcal{M}_{RREP} = \eta_i (h+1) + \sum_{k_k \in K} (h+1)^{k_k})$ and the time complexity is $T = T_{RREQ} + T_{RREP} = 2(h+1)$ time steps.

3) EXPERIMENTAL RESULTS

This section presents our experimental results. In Section V-B-3-i, we present the results of the effects of learning rate α on the RL scheme. In Section V-B-3-ii, we present the results of the comparison of different route selection schemes, namely Highest-Channel (HC), RL-based, as well as SL-based schemes. In Section V-B-3-iii, we compare the performance of both 6-node and 10-node topologies.

a: EFFECTS OF LEARNING RATE α ON RL-BASED SCHEMES

This section presents the effects of learning rate α on the QoS parameters (i.e. throughput and packet delivery ratio, as well as routing stability) of a RL-based scheme (i.e. TRL approach) in the 6-node topology. As TRL and ARL schemes produce approximately similar results, only results of the TRL scheme are presented (see Section V-B-3-ii). The learning rate α is an important parameter that affects the learning speed. In this work, the learning rate α is dependent and adjusted according to the level of dynamicity of the operating environment. Specifically, higher (or lower) α value is needed for operating environment with higher (or lower) dynamicity. This is shown in Fig. 12 in which higher α value shows greater performance enhancement providing higher throughput compared to lower α value as the PUs' OFF time increases beyond 30 s, with the optimal throughput being achieved with $\alpha = 0.9$.

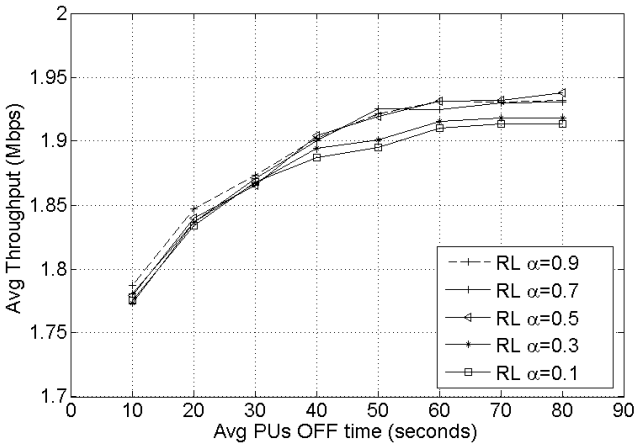

FIGURE 12. Average throughput versus PU-OFF time $\lambda^{P}_{c,OFF}$ at different values of α for TRL scheme using 6-node topology.

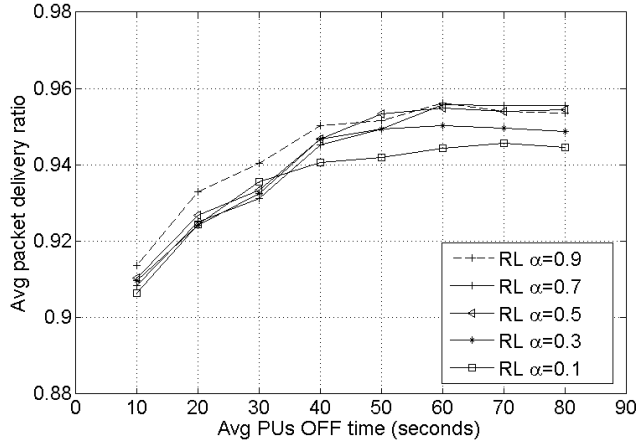

FIGURE 13. Average packet delivery ratio versus PU-OFF time $\lambda^{P}_{c,OFF}$ at different values of α for TRL scheme using 6-node topology.

Fig. 12 and Fig. 13 show that the throughput and packet delivery ratio slightly increase (note that the y-axis starts at 1.7 Mbps and 0.88 in the figures, respectively) with

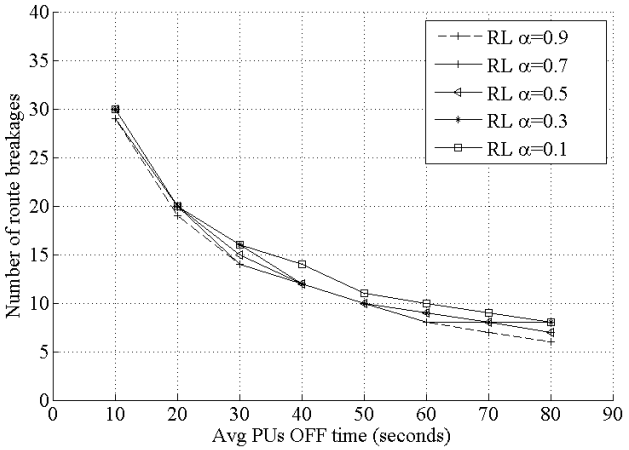

FIGURE 14. Average number of route breakages versus PU-OFF time $\lambda^{P}_{c,OFF}$ at different values of α for TRL scheme using 6-node topology.

increasing learning rate (i.e. $\alpha \geq 0.5$). Similarly, Fig. 14 shows that the number of route breakages is lesser when the learning rate is higher (i.e. $\alpha \geq 0.5$). This is because, as α increases, the RL-based scheme is more dependent on the current knowledge (i.e., $\Gamma^{k_k}_{\beta,t+1}$) due to the high temporal variability of the wireless channels (i.e., channel available time), rather than the previous knowledge (i.e., $Q^{m_{1,k_k}}_{t}(s_t, a^{m_{n_1,j_1}}_{t})$). With learning rate $\alpha = 0.9$, the RL approach provides the best possible network performance, and so this value is chosen for comparison with the other approaches in Section V-B-3-ii, as well as comparison in the performance achieved by both 6-node and 10-node topologies in Section V-B-3-iii.

b: COMPARISON OF ROUTE SELECTION SCHEMES

This section presents the experimental results of the RL-based schemes (see Section IV-C) and SL-based scheme (see Section IV-D) in the 6-node and 10-node topologies. We compare the results with Highest-Channel (HC), which is a RL-based scheme that selects the route in a multi-hop CR networks with the highest number of available channels [46], instead of highest channel available time. The main objective of the proposed schemes is to select the best possible route from a source node to a destination node in a multi-hop CR networks in order to improve QoS parameters, particularly throughput and packet delivery ratio, as well as the number of route breakages which affects the routing stability. Fig. 15 and Fig. 16 show that throughput and packet delivery ratio performance increase with the average PUs' OFF time from 10 s to 50 s in the 6-node topology and stabilize when the average PUs' OFF time reaches approximately 50s. Fig. 17 shows that the number of route breakages reduces with increasing average PUs' OFF time and stabilizes when the PUs' OFF time reaches 60s. Similarly, Fig. 18 and Fig. 19 show that throughput and packet delivery ratio performance increase with the average PUs' OFF time from 10 s to 50 s in the 10-node topology and stabilizes when the average PUs' OFF time reaches approximately 50 s. Fig. 20 shows

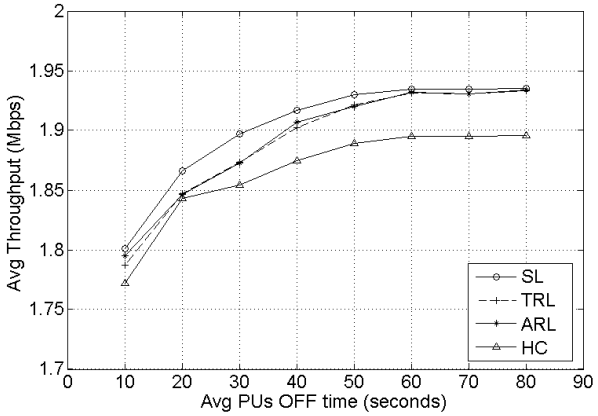

FIGURE 15. Average throughput versus PU-OFF time $\lambda^{P}_{c,OFF}$ for a 6-node topology.

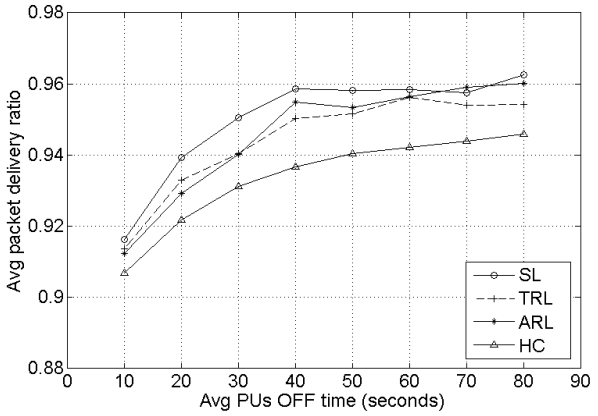

FIGURE 16. Average packet delivery ratio versus PU-OFF time $\lambda^{P}_{c,OFF}$ for a 6-node topology.

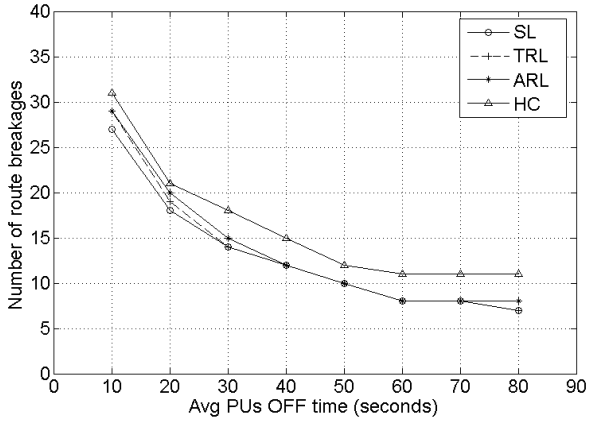

FIGURE 17. Average number of route breakages versus PU-OFF time $\lambda^{P}_{c,OFF}$ for a 6-node topology.

that the number of route breakages reduces with increasing average PUs' OFF time and stabilizes when the PUs' OFF time reaches 60 s.

Overall, the SL-based scheme achieves higher throughput and packet delivery ratio, as well as lower number of route breakages, in comparison with the RL-based and

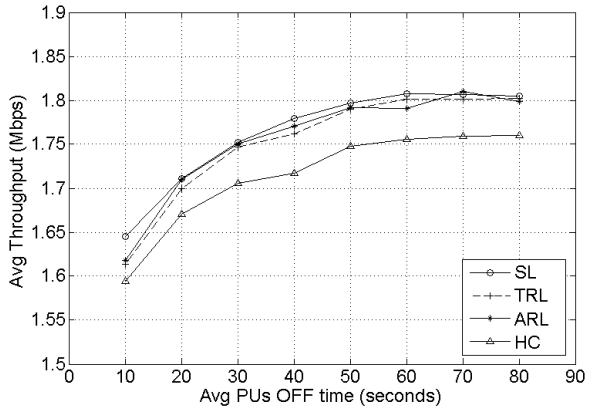

FIGURE 18. Average throughput versus PU-OFF time $\lambda^{P}_{c,OFF}$ for a 10-node topology.

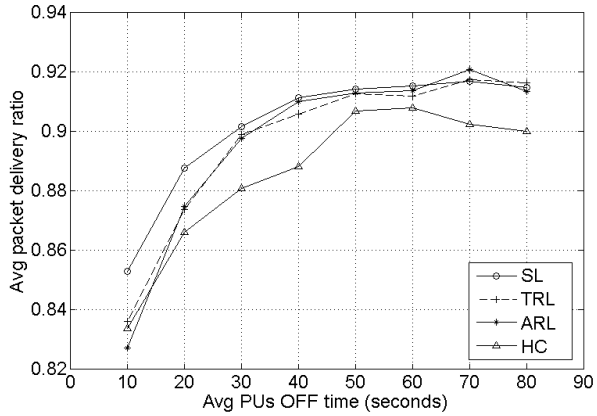

FIGURE 19. Average packet delivery ratio versus PU-OFF time $\lambda^{P}_{c,OFF}$ for a 10-node topology.

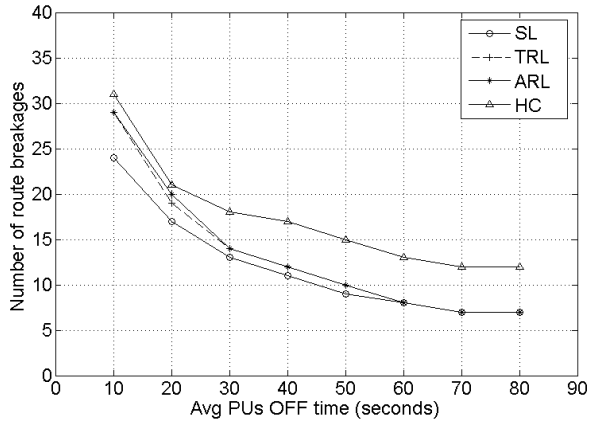

FIGURE 20. Average number of route breakages versus PU-OFF time $\lambda^{P}_{c,OFF}$ for a 10-node topology.

HC schemes. This is because the SL-based scheme, SUs are aware of the exact channel available time (or PUs' activities) as PUs share their respective spectrum occupancy map (i.e., $\lambda^{P}_{cc,OFF}$) with the SUs located within their respective transmission range. Although the RL-based and HC schemes receive ideal sensing outcomes, their performance degrades in comparison to the SL approach due to the channel sensing delay. The number of route breakages of the SL-based and

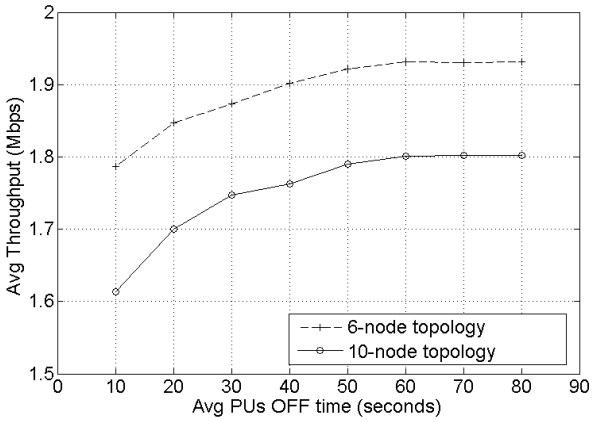

FIGURE 21. Average throughput versus PU-OFF time $\lambda^{P}_{c,OFF}$ at $\alpha = 0.9$ for TRL scheme in 6-node and 10-node topologies.

RL-based schemes are lower than that of the HC scheme. In all cases, the HC scheme shows the least performance in comparison with SL-based and RL-based schemes as it selects a route with the highest number of available channels, which may have low channel available time at its bottleneck link. In addition, the ARL scheme shows a very minor improvement in comparison with the TRL scheme for both 6-node and 10-node topologies. The RL schemes are primarily dependent on the estimated channel available time in order to compute the Q-value. However, the key difference between the TRL scheme and the ARL scheme is that the ARL scheme further consider the average Q-value for the computation of Q-value (see Section IV-C); which determines the stability in terms of the route(s) selection in the past.

Next, we compare the QoS performance achieved by the 6-node and 10-node topologies in order to investigate the scalability aspect of the schemes. The maximum throughput achieved by the 6-node and 10-node topologies are 1.94 Mbps and 1.8 Mbps as shown in Fig. 15 and Fig. 18, respectively. The maximum packet delivery ratio achieved by the 6-node and 10-node topologies are 96% and 92% as shown in Fig. 16 and Fig. 19, respectively. In short, the throughput and packet delivery ratio deteriorate in the 10-node topology with increasing number of nodes due to the increasing number of hops in a route resulting in higher packet loss. Similarly, we compare the routing stability achieved by the 6-node and 10-node topologies. The number of route breakages is approximately similar in both 6-node and 10-node topologies, and the maximum number of route breakages is 31 in each case as shown in Fig. 17 and Fig. 20. This is because a route breakage is dependent on the channel available time at the bottleneck link, which occurs similarly in both topologies.

c: COMPARISON OF PERFORMANCE BETWEEN 6-NODE AND 10-NODE TOPOLOGIES

This section compares the QoS performance (i.e., throughput and packet delivery ratio, as well as routing stability) achieved by TRL in 6-node and 10-node topologies. In general, the 6-node topology provides better network performance

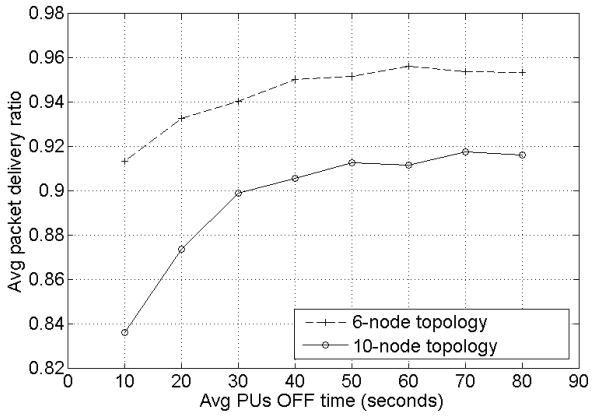

FIGURE 22. Average packet delivery ratio versus PU-OFF time $\lambda^{P}_{c,OFF}$ at $\alpha = 0.9$ for TRL scheme in 6-node and 10-node topologies.

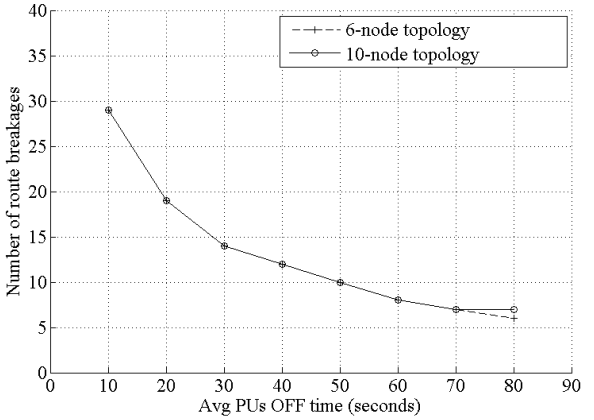

FIGURE 23. Average number of route breakages versus PU-OFF time $\lambda^{P}_{c,OFF}$ at $\alpha = 0.9$ for TRL scheme in 6-node and 10-node topologies.

compared to the 10-node topology. The maximum throughput achieved by the 6-node and 10-node topologies are 1.93 Mbps and 1.8 Mbps respectively, as shown in Fig. 21. The maximum packet delivery ratio achieved by the 6-node and 10-node topologies are 95.8% and 91.9% respectively, as shown in Fig. 22. The performance deteriorates in the 10-node topology due to the fact that it has higher number of hops in a route resulting in higher packet loss as compared to 6-node topology. The number of route breakages are approximately similar in both topologies because of the same PUs activity level (i.e., $\lambda^{p}_{c_c,OFF} = \{10, 20, 30, 40, 50, 60, 70, 80\}$ s and $\lambda^{p}_{c_c,OFF} = 15$ s) as shown in Fig. 23.

VI. CONCLUSION AND FUTURE WORK

In this article, we have proposed and implemented three route selection schemes on a USRP/GNU Radio testbed environment based on reinforcement learning (RL) and spectrum leasing (SL) approaches in order to enhance the network performance of multi-hop cognitive radio (CR) networks. The three schemes are the traditional RL (TRL) approach, the RL approach with average Q-value (ARL), and a SL approach. The RL-based schemes use an artificial intelligence technique to make route selection; whereas, the SL-based scheme uses a spectrum occupancy map received from primary users (PUs) to make route selection. Experimental results show that the proposed RL-based and SL-based schemes aim to select routes with the highest Q-value and highest minimum channel available time, respectively, contributing to a lower number of route breakages and higher throughput and packet delivery ratio compared to an existing route selection scheme called highest-channel (HC). The reason is that, in the RL-based schemes, intelligence is incorporated to find the estimated value of channel available time at the bottleneck link. Similarly, in the SL-based scheme, the SUs are aware of the PUs' activities and so the exact channel available time at the bottleneck link is used. Whereas, in the HC scheme, routes are selected on the basis of the highest number of available channels irrespective of the channel available time. The proposed RL-based and SL-based schemes appear to perform well as they provide enhanced performance in both 6-node and 10-node topologies. Our results show that, the 6-node topology achieves better performance than the 10-node topology. Although the 10-node topology has higher number of routes than the 6-node topology, the degradation in the performance is due to the higher number of hops in the 10-node topology. In this article, we have also mathematically analyzed the complexity of our route selection schemes, and found that the complexity increases with the number of routes and their respective number of hops, which deteriorate the overall QoS performance (i.e., throughput and packet delivery ratio).

In the future, we plan to implement a cross-layer design that enables the physical, data link and network layers to optimize network performance and to relax the assumptions made in this article to provide a more realistic test environment. The cross-layer design enables: (a) the channel sensing mechanism in the data link layer to provide channel sensing outcomes to the route selection mechanism in the network layer, (b) the channel access mechanism in the data link layer to allow SUs to share the available channels of a single channel among themselves, and (c) the transmission and reception mechanisms in the physical layer to address the typical phenomena, such as fading and shadowing, in the presence of dynamicity of PUs' activities which is the main characteristic of CR such that the USRP SU nodes are not placed close to each other. We also plan to test our cross-layer design in a larger network with more USRP SU nodes to provide a more realistic test environment in which there are higher number of routes, as well as higher number of hops in each route. More extensions to the RL approaches applied in this work, such as the multi-agent approach, is also part of the future work to improve the SUs' network performance in more complex and realistic scenarios.

REFERENCES

[1] Z. Chen and C. Chen, "Adaptive energy-efficient spectrum probing in cognitive radio networks," *Ad Hoc Netw.*, vol. 13, pp. 256–270, Feb. 2014.
[2] Q. Liang, X. Wang, X. Tian, F. Wu, and Q. Zhang, "Two-dimensional route switching in cognitive radio networks: A game-theoretical framework," *IEEE/ACM Trans. Netw.*, vol. 23, no. 4, pp. 1053–1066, Aug. 2014.

[3] D. Xue and E. Ekici, "Cross-layer scheduling for cooperative multi-hop cognitive radio networks," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 3, pp. 534–543, Mar. 2013.

[4] A. Bourdena, X. C. Mavromoustakis, G. Kormentzas, E. Pallis, G. Mastorakis, and B. M. Yassein, "A resource intensive traffic-aware scheme using energy-aware routing in cognitive radio networks," *Future Generat. Comput. Syst.*, vol. 39, pp. 16–28, Oct. 2014.

[5] J. Backens, C. Xin, and M. Song, "A novel protocol for transparent and simultaneous spectrum access between the secondary user and the primary user in cognitive radio networks," *Comput. Commun.*, vol. 69, pp. 98–106, Sep. 2015.

[6] S. L. Cardoso *et al.*, "CorteXlab: A cognitive radio testbed for reproducible experiments," in *Proc. Wireless, Virginia Tech Symp.* Blacksburg, VA, USA, 2014, pp. 1–7.

[7] V. L. Nir and B. Scheers, "Implementation of an adaptive OFDMA PHY/MAC on USRP platforms for a cognitive tactical radio network," in *Proc. Military Commun. Inf. Syst. Conf. (MCC)*, Gdansk, Poland, Oct. 2012, pp. 1–7.

[8] Z. Yan, Z. Ma, H. Cao, G. Li, and W. Wang, "Spectrum sensing, access and coexistence testbed for cognitive radio using USRP," in *Proc. Int. Conf. Circuits Syst. Commun. (ICCSC)*, Shanghai, China, May 2008, pp. 270–274.

[9] L. Yang, Z. Zhang, W. Hou, B. Y. Zhao, and H. Zheng, "Papyrus: A software platform for distributed dynamic spectrum sharing using SDRs," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 41, no. 1, pp. 31–47, Jan. 2011.

[10] Q. Zhang, J. Jia, and J. Zhang, "Cooperative relay to improve diversity in cognitive radio networks," *IEEE Commun. Mag.*, vol. 47, no. 2, pp. 111–117, Feb. 2009.

[11] A. Puschmann, M. A. Kalil, and A. Mitschele-Thiel, "Implementation and evaluation of a practical SDR testbed," in *Proc. 4th Int. Conf. Cognit. Radio Adv. Spectr. Manage. (CogART)*, Barcelona, Spain, 2011, Art. no. 15.

[12] L. Sun, W. Zheng, N. Rawat, V. Sawant, and D. Koutsonikolas, "Performance comparison of routing protocols for cognitive radio networks," *IEEE Trans. Mobile Comput.*, vol. 14, no. 6, pp. 1272–1286, Jun. 2014.

[13] X. Huang, D. Lu, P. Li, and Y. Fang, "Coolest path: Spectrum mobility aware routing metrics in cognitive ad hoc networks," in *Proc. 31st Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Minneapolis, MN, USA, Jun. 2011, pp. 182–191.

[14] B. P. Nagaraju, L. Ding, T. Melodia, N. S. Batalama, A. D. Pados, and J. D. Matyjas, "Implementation of a distributed joint routing and dynamic spectrum allocation algorithm on USRP2 radios," in *Proc. 7th Commun. Soc. Conf. Sensor Mesh Ad Hoc Commun. Netw. (SECON)*, Boston, MA, USA, Jun. 2010, pp. 1–2.

[15] Y. Huang, P. A. Walsh, Y. Li, and S. Mao, "A distributed polling service-based MAC protocol testbed," *Int. J. Commun. Syst.*, vol. 27, no. 12, pp. 3901–3921, Dec. 2014.

[16] Z. Chen, N. Guo, Z. Hu, and C. R. Qiu, "Experimental validation of channel state prediction considering delays in practical cognitive radio," *IEEE Trans. Veh. Technol.*, vol. 60, no. 4, pp. 1314–1325, May 2011.

[17] M. El-Hajjar, A. Q. Nguyen, R. G. Maunder, and S. X. Ng, "Demonstrating the practical challenges of wireless communications using USRP," *IEEE Commun. Mag.*, vol. 52, no. 5, pp. 194–201, May 2014.

[18] A. Briand, B. B. Albert, and C. E. Gurjao, "Complete software defined RFID system using GNU radio," in *Proc. Int. Conf. RFID-Technol. Appl. (RFID-TA)*, Nice, France, Nov. 2012, pp. 287–291.

[19] M. Cesana, F. Cuomo, and E. Ekici, "Routing in cognitive radio networks: Challenges and solutions," *Ad Hoc Netw.*, vol. 9, no. 3, pp. 228–248, May 2012.

[20] S. Bayhan and F. Alagoz, "Scheduling in centralized cognitive radio networks for energy efficiency," *IEEE Trans. Veh. Technol.*, vol. 62, no. 2, pp. 582–595, Feb. 2013.

[21] A. R. Syed and K.-L. A. Yau, "Spectrum leasing in cognitive radio networks: A survey," *Int. J. Distrib. Sensor Netw.*, vol. 2014, Feb. 2014, Art. no. 329235.

[22] A. V. Kordali and P. G. Cottis, "A contract-based spectrum trading scheme for cognitive radio networks enabling hybrid access," *IEEE Access*, vol. 3, pp. 1531–1540, Jul. 2015.

[23] M. D. Felice, K. R. Chowdhury, W. Kim, A. Kassler, and L. Bononi, "End-to-end protocols for cognitive radio ad hoc networks: An evaluation study," *Perform. Eval.*, vol. 68, no. 9, pp. 859–875, Sep. 2011.

[24] D. P. Patil and V. M. Wadhai, "NS2 based advanced routing model for cognitive radio networks from dynamic spectrum management perception," in *Proc. Students, Conf. Elect., Electron. Comput. Sci. (SCEECS)*, Bhopal, Mar. 2014, pp. 1–5.

[25] D. Chiarotto, O. Simeone, and M. Zorzi, "Spectrum leasing via cooperative opportunistic routing techniques," *IEEE Trans. Wireless Commun.*, vol. 10, no. 9, pp. 2960–2970, Sep. 2011.

[26] J. Qadir, "Artificial intelligence based cognitive routing for cognitive radio networks," *Artif. Intell. Rev.*, vol. 45, no. 1, pp. 25–96, Jan. 2016.

[27] I. Pefkianakis, S. H. Y. Wong, and S. Lu, "SAMER: Spectrum aware mesh routing in cognitive radio networks," in *Proc. 3rd Symp. New Frontiers Dyn. Spectr. Access Netw. (DySPAN)*, Chicago, IL, USA, Oct. 2008, pp. 1–5.

[28] S. A. Cacciapuoti, M. Caleffi, and L. Paura, "Reactive routing for mobile cognitive radio ad hoc networks," *Ad Hoc Netw.*, vol. 10, no. 5, pp. 803–815, Jul. 2012.

[29] K. C. How, M. Ma, and Y. Qin, "Routing and QoS provisioning in cognitive radio networks," *Comput. Netw.*, vol. 55, no. 1, pp. 330–342, Jan. 2011.

[30] K. R. Chowdhury and I. F. Akyildiz, "CRP: A routing protocol for cognitive radio ad hoc networks," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 4, pp. 794–804, Apr. 2011.

[31] S. Deng, J. Chen, H. He, and W. Tang, "Collaborative strategy for route and spectrum selection in cognitive radio networks," in *Proc. Int. Conf. Future Generat. Commun. Netw. (FGCN)*, Jeju, South Korea, Dec. 2007, pp. 168–172.

[32] G. M. Zhu, I. F. Akyildiz, and G.-S. Kuo, "STOD-RP: A spectrum-tree based on-demand routing protocol for multi-hop cognitive radio networks," in *Proc. Global Telecommun. Conf. (GLOBECOM)*, New Orleans, LO, USA, Nov./Dec. 2008, pp. 1–5.

[33] M. Rovcanin, E. D. Poorter, I. Moerman, and P. Demeester, "A reinforcement learning based solution for cognitive network cooperation between co-located, heterogeneous wireless sensor networks," *Ad Hoc Netw.*, vol. 17, pp. 98–113, Jun. 2014.

[34] N. Coutinho *et al.*, "Dynamic dual-reinforcement-learning routing strategies for quality of experience-aware wireless mesh networking," *Comput. Netw.*, vol. 88, pp. 269–285, Sep. 2015.

[35] M. Danieletto, G. Quer, R. R. Rao, and M. Zorzi, "CARMEN: A cognitive networking testbed on android OS devices," *IEEE Commun. Mag.*, vol. 52, no. 9, pp. 98–107, Sep. 2014.

[36] N. B. Truong, Y.-J. Suh, and C. Yu, "Latency analysis in GNU radio/USRP-based software radio platforms," in *Proc. Military Commun. Conf. (MILCOM)*, San Diego, CA, USA, Nov. 2013, pp. 305–310.

[37] R. Chen, J.-M. Park, and K. Bian, "Robust distributed spectrum sensing in cognitive radio networks," in *Proc. 27th Conf. Comput. Commun. (INFOCOM)*, Phoenix, AZ, USA, Apr. 2008, pp. 31–35.

[38] X. Wang, M. Sheng, D. Zhai, J. Li, G. Mao, and Y. Zhang, "Achieving bi-channel-connectivity with topology control in cognitive radio networks," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 11, pp. 2163–2176, Nov. 2014.

[39] H. Sun, A. Nallanathan, S. Cui, and C.-X. Wang, "Cooperative wideband spectrum sensing over fading channels," *IEEE Trans. Veh. Technol.*, vol. 65, no. 3, pp. 1382–1394, Mar. 2016.

[40] I. E. Atawi, O. S. Badarneh, M. S. Aloqlah, and R. Mesleh, "Energy-detection based spectrum-sensing in cognitive radio networks over multipath/shadowed fading channels," in *Proc. Wireless Telecommun. Symp. (WTS)*, Apr. 2015, pp. 1–6.

[41] M. H. Rehmani, C. A. Viana, H. Khalife, and S. Fdida, "SURF: A distributed channel selection strategy for data dissemination in multi-hop cognitive radio networks," *Comput. Commun.*, vol. 36, nos. 10–11, pp. 1172–1185, Jun. 2013.

[42] A. J. Boyan and L. M. Littman, "Packet routing in dynamically changing networks: A reinforcement learning approach," in *Proc. Adv. Neural Inf. Process. Syst.*, San Francisco, CA, USA, 1994, p. 671.

[43] H. A. A. Al-Rawi, K.-L. A. Yau, H. Mohamad, N. Ramli, and W. Hashim, "Reinforcement learning for routing in cognitive radio ad hoc networks," *Sci. World J.*, vol. 2014, 2014, Art. no. 960584.

[44] R. Arroyo-Valles, R. Alaiz-Rodríguez, A. Guerrero-Curieses, and J. Cid-Sueiro, "Q-probabilistic routing in wireless sensor networks," in *Proc. 3rd Int. Conf. Intell. Sensors Netw. Inf. (ISSNIP)*, Dec. 2007, pp. 1–6.

[45] (2015). *VERT900 Antenna*. [Online]. Available: http://www.ettus.com/product/details/VERT900

[46] B. Xia, H. M. Wahab, Y. Yang, Z. Fan, and M. Sooriyabandara, "Reinforcement learning based spectrum-aware routing in multi-hop cognitive radio networks," in *Proc. 4th Int. Conf. Cognit. Radio Oriented Wireless Netw. Commun. (CROWNCOM)*, Hannover, Germany, Jun. 2009, pp. 1–5.

AQEEL RAZA SYED received the M.S. degree in electronic engineering (communication and signal processing) from Hamdard University, Pakistan, in 2006, and the M.Sc. degree in electrical engineering (telecommunication) from the Blekinge Institute of Technology, Sweden, in 2009. He is currently pursuing the Ph.D. degree with Cognitive Radio Network, Sunway University, Malaysia, under the supervision of Dr. K. L. A. Yau. His research interests include cognitive radio networks, multihop communication, reinforcement learning, and real test bed implementation. He has professional industry experience in networks and telecommunication in renowned organizations in Pakistan and Denmark.

KOK-LIM ALVIN YAU received the B.Eng. degree (Hons.) in electrical and electronics engineering from Universiti Teknologi PETRONAS, Malaysia, in 2005, the M.Sc. degree in electrical engineering from the National University of Singapore in 2007, and the Ph.D. degree in network engineering from the Victoria University of Wellington, New Zealand, in 2010. He is currently an Associate Professor with the Department of Computing and Information Systems, Sunway University. He was a recipient of the 2007 Professional Engineer Board of Singapore Gold Medal for being the best graduate of the M.Sc. degree in 2006/2007. He is a Researcher, a Lecturer, and a Consultant in cognitive radio, wireless networks applied artificial intelligence, and reinforcement learning. He serves as an Editor of the *KSII Transactions on Internet and Information Systems*, a Guest Editor of the Special Issues of the IEEE Access and the *IET Networks*, and a regular Reviewer for over 20 journals, including the IEEE journals and magazines, the IEEEAd Hoc Networks, the *IET Communications*, and others. He serves as a TPC Member and a Reviewer for various major international conferences, including ICC, VTC, LCN, Globecom, and AINA. He also served as the General Co-Chair of the IET ICFCNA'14 and the Co-Chair of the Organizing Committee of the IET ICWCA'12.

JUNAID QADIR (SM'–14) received the B.S. degree in electrical engineering from UET, Lahore, Pakistan, and the Ph.D. degree from the University of New South Wales, Australia, in 2008. He was an Assistant Professor with the School of Electrical Engineering and Computer Sciences (SEECS), National University of Sciences and Technology (NUST), Pakistan. He was with the Cognet Laboratory, SEECS, with a focus on cognitive networking and the application of computational intelligence techniques in networking. He is currently an Associate Professor with the Information Technology University of the Punjab (ITU), Lahore, Pakistan. He is also the Director of the IHSAN Laboratory, ITU, with a focus on deploying ICT for development, and is involved in systems and networking research. His research interests include the application of algorithmic, machine learning, and optimization techniques in networks. In particular, he is interested in the broad areas of wireless networks, cognitive networking, software-defined networks, and cloud computing. He is a member of the ACM. He received the Highest National Teaching Award in Pakistan-the Higher Education Commission's Best University Teacher Award-from 2012 to 2013. He has been nominated for this award twice (in 2011, and from 2012 to 2013). He is a regular Reviewer for a number of journals and has served in the program committee of a number of international conferences. He serves as an Associate Editor of the IEEE Access, the *IEEE Communication Magazine*, and *Nature Big Data Analytics* (Springer). He was the Lead Guest Editor of the Special Issue of the Artificial Intelligence Enabled Networking in the IEEE Access and the feature topic of the Wireless Technologies for Development in the *IEEE Communications Magazine*.

HAFIZAL MOHAMAD received the B.Eng. degree (Hons.) and the Ph.D. degree in electronic engineering from the University of Southampton, U.K., in 1998 and 2003, respectively.

From 1998 to 2007, he was a Faculty Member with Multimedia University, Malaysia. He served a short stint as a Visiting Fellow with NICT, Yokosuka, Japan, in 2005. He is currently a Senior Staff Researcher with Wireless Innovation, MIMOS Berhad, where he leads a team, which is involved in cognitive radio and mesh network. He holds seven patents granted and 34 patents filed. He has authored over 90 journals and conference papers.

NORDIN RAMLI received the B.Eng. degree in electrical engineering from Keio University, Japan, in 1999, and the M.Eng. and Ph.D. degrees in electronics engineering from The University of Electro-Communications, Japan, in 2005 and 2008, respectively. He is currently a Staff Researcher with the Wireless Network and Protocol Research, MIMOS Berhad, Malaysia. He holds more than 20 patents. He has authored over 60 journals and conference papers. His current research and development interests are in the area of cognitive radio, TV white space, space-time processing, equalization, adaptive array system, and wireless mesh networking.

SYE LOONG KEOH received the Ph.D. degree in computing science from Imperial College London in 2005. He was a Post-Doctoral Research Associate with Imperial College London from 2005 to 2008. He was a Senior Scientist with Philips Research Eindhoven from 2008 to 2013, where he was involved in standardizing Digital Rights Management technology for content protection. He was also involved in projects related to medical security and security protocols design for Philips lighting systems. He is currently an Assistant Professor with the School of Computing Science, University of Glasgow, Singapore.

He has been actively contributing to the standardization of security protocols for Internet of Things in the Internet Engineering Task Force since 2011, specifically he contributes to the Constrained Restful Environment, DTLS in Constrained Environment, and Lightweight Implementation Guidance working groups. His research interests include policy-based management, building and home automation, security protocol design, security and trust management, Internet of Things, and pervasive computing.