Digital Communications and Networks 9 (2023) 367–382

Contents lists available at ScienceDirect

Digital Communications and Networks

journal homepage: www.keaipublishing.com/dcan

Dynamic channel estimation-aware routing protocol in mobile cognitive radio networks for smart IIoT applications

Qusay M. Salih [a], Md Arafatur Rahman [b,*], A. Taufiq Asyhari [c], Muhammad Kamran Naeem [b], Mohammad Patwary [b], Ryan Alturki [d], Mohammed Abdulaziz Ikram [e]

[a] Faculty of Computing, Universiti Malaysia Pahang, 26300, Gambang, Pahang, Malaysia
[b] School of Engineering, Computing and Mathematical Sciences, University of Wolverhampton, UK
[c] Data Science Program, Monash University, Indonesia, and Birmingham City University, B4 7XG, UK
[d] Department of Information Science, College of Computer and Information Systems, Umm Al-Qura University, Saudi Arabia
[e] Computer Science Department, Al Jumoum University College, Umm Al-Qura University, Saudi Arabia

ARTICLE INFO

Keywords:
Channel selection
Cross-layer design
Mobile cognitive radio networks
Routing protocol
IIoT applications

ABSTRACT

Cognitive Radio Networks (CRNs) have become a successful platform in recent years for a diverse range of future systems, in particularly, industrial internet of things (IIoT) applications. In order to provide an efficient connection among IIoT devices, CRNs enhance spectrum utilization by using licensed spectrum. However, the routing protocol in these networks is considered one of the main problems due to node mobility and time-variant channel selection. Specifically, the channel selection for routing protocol is indispensable in CRNs to provide an adequate adaptation to the Primary User (PU) activity and create a robust routing path. This study aims to construct a robust routing path by minimizing PU interference and routing delay to maximize throughput within the IIoT domain. Thus, a generic routing framework from a cross-layer perspective is investigated that intends to share the information resources by exploiting a recently proposed method, namely, Channel Availability Probability. Moreover, a novel cross-layer-oriented routing protocol is proposed by using a time-variant channel estimation technique. This protocol combines lower layer (Physical layer and Data Link layer) sensing that is derived from the channel estimation model. Also, it periodically updates and stores the routing table for optimal route decision-making. Moreover, in order to achieve higher throughput and lower delay, a new routing metric is presented. To evaluate the performance of the proposed protocol, network simulations have been conducted and also compared to the widely used routing protocols, as a benchmark. The simulation results of different routing scenarios demonstrate that our proposed solution outperforms the existing protocols in terms of the standard network performance metrics involving packet delivery ratio (with an improved margin of around 5–20% approximately) under varying numbers of PUs and cognitive users in Mobile Cognitive Radio Networks (MCRNs). Moreover, the cross-layer routing protocol successfully achieves high routing performance in finding a robust route, selecting the high channel stability, and reducing the probability of PU interference for continued communication.

1. Introduction

1.1. Background

Cognitive Radio Networks (CRNs) leverage the adaptive mechanism to effectively utilize an existing underutilized wireless spectrum. Industrial Internet of Things (IIoT) applications require large number of sensors and actuators need to be communicated where spectrum scarcity will be the key challenge. The Cognitive Radio (CR) presents a contrasting paradigm in the communication lower layer operations and ultimately impacts the upper layers' functionalities. Despite their complications, CRNs have attracted a wide range of interests in the past decades. Attractive main propositions include their capability to accommodate potentially many unlicensed or commonly known as Cognitive Users (CUs) to use licensed spectrum bands opportunistically and thereby assist in fully exploiting the use of a licensed spectrum more

* Corresponding author.
E-mail addresses: qusaysalih81@gmail.com (Q.M. Salih), arafatur.rahman@ieee.org, arafatur.rahman@wlv.ac.uk (M.A. Rahman), taufiq-a@ieee.org (A.T. Asyhari), m.k.naeem@wlv.ac.uk (M.K. Naeem), patwary@wlv.ac.uk (M. Patwary), rmturki@uqu.edu.sa (R. Alturki), maikram@uqu.edu.sa (M.A. Ikram).

https://doi.org/10.1016/j.dcan.2023.01.019
Received 18 October 2021; Received in revised form 31 October 2022; Accepted 29 January 2023
Available online 3 February 2023

efficiently [1,2]. Therefore, CRNs are expected to increase spectrum usage, which mitigates the spectrum scarcity issue introduced by the IIoT applications. However, careful transmission synchronization design for PUs and CUs is needed prior to deployment especially in the IIoT. It is expected that CUs must prevent causing interference and interrupting the PUs transmission by adaptively switching to another unused channel because PUs have a higher priority for using the licensed spectrum band [3]. This is achieved by using Dynamic Spectrum Access (DSA), which enables CUs to switch among different spectrum holes.

The CRNs routing protocol has received much attention in recent years due to using the idle licensed spectrum to send and receive the data packet between the pair of CUs, which offers the desired benefits for numerous potential applications, such as IIoT, smart city, military, healthcare and self-driving cars. For example, in smart city applications, this protocol will offer more spectrum for data-packets send/receive, enhance the communication of the smart applications, and reduce the economic cost for the users [4–6]. As a result, the interest in large-scale CRNs for different applications and the development of routing protocols has been growing recently. There are several routing protocol studies on CRNs have been presented in the literature to reduce PU interference by detecting free (idle) channels and improving routing performance [7–12]. However, little attention has been paid to maximizing routing reliability considering the impact of time-variant channel estimation and node mobility. The primary challenge for routing protocols in CRNs is setting up and selecting a robust-stable channel route over multi-channel/multi-path between the source node and destination with high routing performance, high channel stability, and lower probability of interference. Therefore, there is a need for CRNs based routing protocol that address these specific challenges.

The routing protocol must gain access to the idle spectrum while avoiding interference from PU transmission operations. Moreover, channel estimation and selection are very challenging that can significantly impact routing performance. The channel estimation suffers from node mobility due to increased interference with the PU and as a result increases the transmission cost. In order to accurately estimate and transmit information over the best idle channel for MCRNs, it is important to consider the effects of time variance on channel estimation. Practically, once the channel selection selects an idle channel frequency with a long holding time of the PU activity, i.e., high channel availability, high quality, and connectivity with neighboring nodes, it can lead to a stable data-packet route, and long and reliable connection time [13]. The CRNs routing protocol, on the other hand, considers issues such as selecting the path with the lowest delay to improve network throughput.

Therefore, creating a robust and stable path in CRNs is considered a difficult target, especially when CUs and/or PUs are mobile [14,15]. However, it may be possible reasonably to construct a stable routing path if the channels picked for connection among CUs in the corresponding route are likely not be swayed by the PUs activity [16,17]. According to authors' knowledge, there is a lack of considerations on estimating and selecting the high channel availability from a routing perspective. In Refs. [18,19], authors' presented routing solutions for more routing stability and high-link quality to increase the average routing throughput, however, when a random channel selection is employed, it resulted in poor performance of the network.

1.2. Motivation and contribution

In MCRNs, the routing path may be affected by the incorrect detection of the occupied channels due to node mobility. Moreover, estimation of channel availability is highly time-variant due to variable spectrum accessibility. Consequently, any misleading decision by the routing protocol may leads to increased control packets resulting in degradation of routing performance and consumption in network resources.

In this study, the main aim is to overcome the potential degradation of the routing performance in MCRNs within IIoT environment. A cross-layer-oriented framework is developed to extend routing functions and methods for implementing and benefiting from lower-layer feedback. In order to mitigate the risk of longer end-to-end delays and reduce network control overhead, a probabilistic metric is also presented for determining the best route to the destination with the least amount of delay and the highest throughput. Moreover, a reliable cross-layer Software-Defined Routing Protocol (SDRP) is also developed to exploit time-variant channel estimation technique which selects the best path using the routing decision engine that can track the adverse impact of PU and CU mobility. In particular, the CUs (sensors and actuators) in IIoT environment the mobility issues play an important role where such time variant mechanism is required. Moreover, the proposed SDRP joins the lower layers (physical layer and data-link layer) resources in the pattern of CAP that emerges from the channel estimation model, which updates every τ and keeps in the routing table for the appropriate route-decision making. Therefore, the proposed protocol provides solutions for multi-channel selection and channel scheduling and increases throughput by setting up a long valid path with lower delay for IIoT enabled network.

The SDRP is implemented in NS-2 for performance evaluation and compared with widely used CRN routing protocols such as Dual Diversity Cognitive Ad-hoc Routing Protocol (D^2CARP) and Cognitive Ad-hoc On-demand Distance Vector (CAODV) [7,18]. The proposed protocol successfully achieves high routing performance to find a robust and reliable route, select a channel with high stability, and reduce the probability of interference with PU for continued communication. The main contributions of this paper can be summarized as follows:

- A cross-layer framework is presented to create a scalable routing decision engine. The proposed framework will extend routing protocol functionality and expand routing table information in CRNs environments. The new reliable cross-layer routing protocol namely Software-Defined Routing Protocol (SDRP) is presented to find the best idle license channel and the associated optimal forwarding path with higher throughput.
- A new routing metric is also designed to identify the path(s) with the lower delay and higher throughput.
- Finally, to evaluate the performance of the proposed framework, a simulation platform is developed. Extensive and diverse simulation analysis have been presented to prove that SDRP significantly improves the routing performance by increasing packet delivery ratio, throughput and reducing delay and overhead, making it a suitable protocol.

The rest of the paper is organized as follows: The literature review is elaborated in Section 2; the problem formulation along with problem statement is described in Section 3; the proposed routing protocol is presented in Section 4; performance analysis of the proposed routing protocol has been presented in Section 5 along with the comparison of existing benchmark routing protocols to evaluate the performance of the proposed framework followed by conclusion in Section 6. Also, we provided the list of frequently-used acronyms in Table 1 and the list of mathematical symbols in Table 2.

Table 1
List of acronyms and corresponding definitions.

Acronyms	Definitions
CRNs	Cognitive Radio Networks
CU	Cognitive User
PUs	Primary Users
CAP	Channel Availability Probability
CQ	Channel Quality
SCRNs	Static Cognitive Radio Networks
MCRNs	Mobile Cognitive Radio Networks
RWPM	Random WayPoint Mobility
RUP	Routing Update Period
PT	Path Throughput
OSI	Open Systems Interconnection

Table 2
Key mathematical symbols.

Symbols	Explanation
ψ_{ij}^{m}	Link-throughput
ψ^{m}	Idle link throughput
$u_i(t)$, $u_j(t)$,	Position of an arbitrary *i-th* CU and *j-th* CU at time t
$v_l(t)$	Position of an arbitrary *l-th* PU at time t
p_{off}^{m}	The probability that PU is inactive on channel m
p_{il}^{m}	The probability that at time t the channel m is available for the transmission of the *i-th*CU without causing interference to the communication of the *l-th* PU
R_{il}	Critical range of *l-th* PU
$d_{il}(t)$	Distance between *i-th* CU and *l-th* PU at time instant t
$\tilde{p}_{ij,m,\mathcal{L}}(T_k)$	the *i-th* CU estimates the probability $p_{ij,m,l}(Tk)$ of channel m that is licensed to a set of PUs $\mathcal{L}$ being available for a pair of *i-th* and *j-th* CUs
$\tilde{d}_{il}(t)$	Estimated distance between *i-th* CU and *l-th* PU at time instant t
PT_{p_i}	Path Throughput for Every Path
τ	Period of Routing Updating for Link-channel Selection
$D_{l_{ij}}^{Best}(\tau)$	Expected Link Delay
$D_{P_{s,d}}^{\mathcal{X}}(\tau)$	Expected Path Delay
$D_{p_{s,d}}^{Best}(\tau)$	Expected Best Path Delay

2. Review of existing works

Significant amount of research has been done by the researchers in the existing literature to address channel selection challenges for routing protocols in CRNs based on the cross and non-cross-layer design. As a result, various protocols have been proposed to enhance the routing performance with reliable channel selection. A critical review for existing related works has been presented as summarized below to validate the existing problems.

A Cognitive Ad-hoc On-demand Distance Vector (CAODV) is proposed in Ref. [18], which was developed to address challenges for PU activity and the diverse options for the path and spectrum. Even though CAODV attempts to find the shortest path, it does not consider the channel characteristics' estimation, and therefore, the optimal route is unsettled. A Dual Diversity Cognitive Ad-hoc Routing Protocol (D^2CARP) is presented by the authors in Ref. [7] which is the revised version of Cognitive Ad hoc On demand Distance Vector (CAODV). In order to use spectrum more efficiently in CRNs, (D^2CARP) jointly employs path and spectrum. Combining the path and spectrum diversity enables CUs to communicate their data-packets from different paths and spectrum bands, and allows CUs to avoid PUs activities. However, similar to CAODV, D^2CARP did not discuss the issues of choosing the optimal idle channel that would become the key shortcoming of this routing protocol for CRNs. A Cross-Layer Routing Protocol (CLRP) is investigated in Ref. [17], where the cross-layer design combines features and functionalities of the physical and network layers. The CLRP considers a potential route by computing the route cost assuming probabilistic availability of the route setup channels. Each Cognitive Radio (CR) node senses some selected channels and estimates the probability of their availability ranging between 0 and 1 based on the birth(β)-death(λ) process. However, one of the drawbacks is the assumption by the authors that for a given certain availability probability for the channels that did not sense by CUs.

A Cross Layered opportunistic Routing Protocol (CLORP) is proposed in Ref. [20], where the information gathered by the lower layers is collected for routing decisions. Energy Detection (ED) at a lower layer is employed to identify unoccupied channels, over which the CUs are allowed to send the data. Moreover, estimating the quality of the link is also considered using the Error Vector Magnitude (EVM) at the Media Access Control (MAC) layer. This protocol has a specific routing request, namely Cognitive Radio Route Request (CRRREQ) that captures the available channel's payload. It is discussed by the authors in Ref. [18] that the process of spectrum sensing at the physical layer presented in by the existing studies in the literature is imperfect and often lacks the accuracy to avoid the PU activity. Consequently, spectrum sensing needs considerable improvement to provide high detection accuracy as discussed in Ref. [19].

A routing and channel selection algorithm named as Proposed Stable Routing (PSR) for CRNs is proposed in Ref. [21] which constructs a reliable route from a source to a destination based on the probability of obtaining a stable channel. When the source node needs to transmit data, it will first send a "hello" packet to all neighboring nodes to discover an available channel. Thereafter, it computes the probability of channel availability based on the ON-OFF period of the PU activity where the sensing information is fed and updated at a pre-defined period. It can incur an additional cost and loss of network resources by sending non-information bearing "hello" packet to sense all neighbors' channels. It is discussed by the authors in Ref. [22] that the update period has not been thoroughly examined and optimized, given its influential role in the case of CU mobility. A Cognitive Radio Routing Protocol (CROP) optimized for the three primary radio navigation problems: link consistency, diversity, and fast route maintenance is presented in Ref. [23]. It uses Smart Spectrum Selection (SSS) and Succeeding Hop Selection (SHS) to achieve a good routing performance. As a result, it enables the CROP to select the available spectrum over the relay node in a single process, making route creation a simple process and reducing the routing overhead as well as increasing the throughput rate. In the route discovery phase, the source node broadcasts the Route Request (RREQ) packet over a Common Control Channel (CCC). However, due to the dynamic spectrum access and diversity of operating channels in CRNs, it may not always be reasonable to find one channel for CCC usage.

To compute the link-lifetime for a mobile cognitive radio ad hoc networks, the author in Ref. [24] presented an analytical model which also considers the mobility of CUs and PU activities. Moreover, a joint Stability-based Routing, Link scheduling and Channel assignment (SRLC) algorithm is presented that depended on combining path-stable selection and designates a frequency resource in channel employment approaches. It is claimed by the authors that the proposed algorithm enhance routing performance in terms of end-to-end delay, throughput, and cumulative interference. In contrast, the analytical model has not considered the probability distribution affecting the route lifetime, which may depend on connectivity, PUs' activities and density.

To find a safe route between the source node and the destination, a cross-layer channel-route protocol based on (physical, link and network) layers for ad hoc cognitive radio networks is proposed in Ref. [25]. By considering the channel's statistics, sensing spectrum consumer periodicity, and time-varying channel quality, a novel metric is derived for efficient channel transmission. A probability of maximum successful transmission is also presented as a measurement for the channel to increase the packet transmission rate and reduce the probability of PU interference. It is claimed by the authors that the proposed algorithm achieves better throughput, end-to-end delay, amount of spectrum hand-off, and interference with the PU. However, the authors have assumed that CU with the highest number of idle channels would be selected as the next-hop. It is not always accurate because the node has to choose the next hop's highest time channel availability. In order to resolve the effects of mobility, PU operation, and spectrum sensing inaccuracy for cognitive radio ad hoc networks, a route stability based multi-path Quality of Service (QoS) routing protocol (SMQRP) is proposed in Ref. [9]. The proposed protocol determines the most suitable primary and alternative paths and guarantees that the best primary and alternative channels are used along those paths. In terms of average end-to-end latency, packet drop likelihood, and throughput, the SMQRP showed substantial improvements.

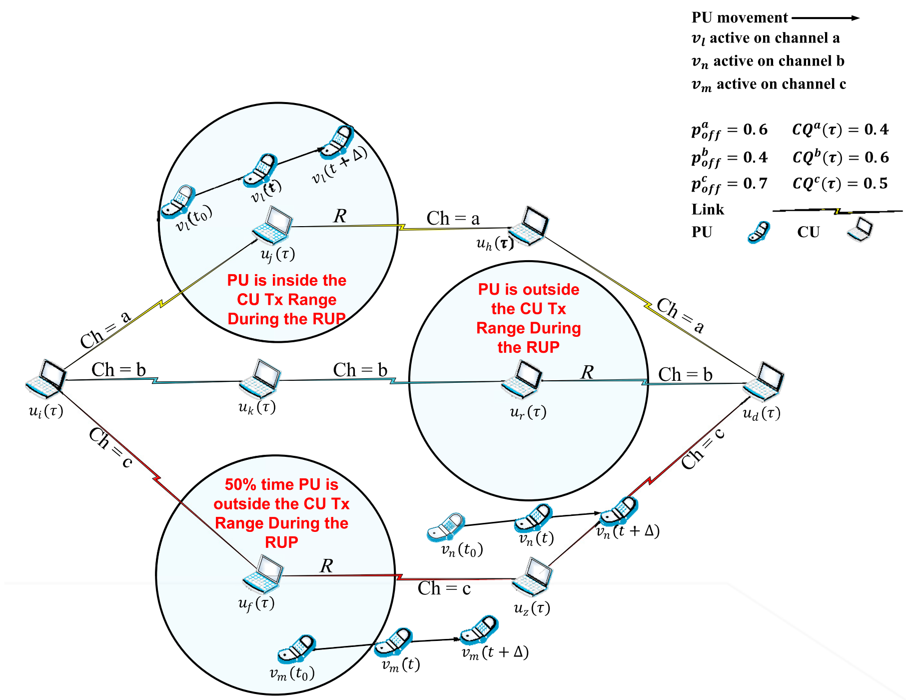

Fig. 1. The impact of Time variant PU and CU relative distance in CRNs.

To select an optimal channel and a distinct route at each next-hop based on the service specifications, a cross-layer routing protocol is presented in Ref. [10]. A mathematical model for predicting the spectrum band, specifically a relation for delay and interference ratio are derived. The proposed framework showed improved throughput, latency, and PU interference. However, this study has overlooked the effect of Secondary User (SU) mobility on the channel-route selection process. A CAODV-based routing approach is proposed in Ref. [11] that employs the multi-path and multi-channel to send the data-packet and deal with the time-variant for PU activity. The aim is to increase the throughput rate by selecting interference free channel. The queuing model for the multi-channel network has been used by Lyapunov optimization, where the Service Price (SP) for per-packet is used as a metric for routing. The proposed CAODV protocol achieved throughput performance, Packet Delivery Ratio (PDR), and reduced delay. However, a drawback to this approach is that the assignment strategies for the channel select only the spectrum available concerning PU interference on the channel, rather than selecting higher channel availability. This study by Ref. [26] proposed the Multi-Adaptive Routing Protocol (MARP) to address the limitations in IoT-based cognitive radio mobile Ad-Hoc networks. MARP was based on a fish's natural behaviour when looking for food. The monitoring board gathers route information from all nodes and synchronises the most up-to-date information while transferring data. In terms of throughput, packet delivery ratio, delay, and energy consumption, the results show that MARP surpasses alternative routing protocols. However, this study has not considered the impact of varying PUs on routing performance.

In [27], this research proposed a Routing in Cognitive Radio Ad Hoc Network (CRAHN) that combined service architecture with eight classes instead of two for CRAHN to manage the access control at PU nodes, reducing SU overhead for various applications. Furthermore, these algorithms choose a route that provides the highest level of QoS and spectrum while disregarding the required QoS level for a specific application, which may be lower than the maximum level. The result demonstrates high reliability, low latency, high throughput rate, and balanced load distribution across all network nodes.

In [28] proposed the cognitive-AODV routing protocol based on preventing channel-route failure and detecting the optimal alternative end-to-end channel-route path between source and destination. According to experimental results, the cognitive AODV routing protocol outperforms current AODV routing cognitive protocols in terms of average end-to-end throughput and latency reduction. However, using the ERROR message always to initiate channel-route find and determine the spectrum handover connection-failure with the different end-to-end channel-route path will consume the network resource, resulting in higher routing overhead.

There are numerous studies discussed in this section that presented routing protocols for CRNs with encouraging results and effective channel selection methods. However, locating a better idle channel is problematic because the time-variant channel estimation and CU mobility compel the routing protocol to update its knowledge about idle channels and combine it with the routing decisions. Moreover, an idle licensed channel is not always represented as an optimal channel to transmit the data over it. Therefore, capturing the optimal channel-route is one of the important considerations when developing an efficient routing protocol. In other words, the studies available in the existing literature do not concentrate on determining high channel availability-quality and mobility from the routing protocol perspective. Therefore, we have presented a Software Defined Radio Protocol (SDRP) that includes setting out, developing and evaluating a new cross-layer routing mechanism exploiting near real-time channel availability and channel quality information based on continuously updated channel characteristics and feeding relevant information to the routing table entries.

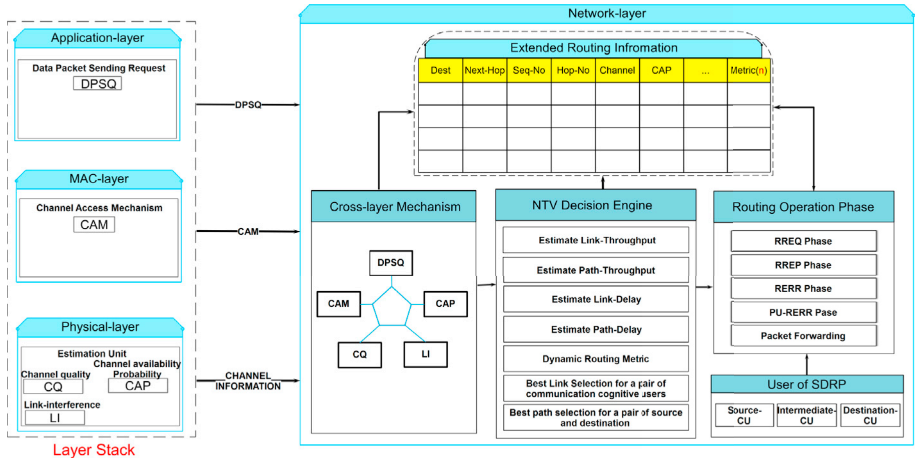

Fig. 2. The Framework of the proposed routing protocol.

3. Problem statement

When a CU requires to communicate with another CU node as shown in Fig. 1, it has to select the path that ensures high quality. In CRNs, numerous routing protocols measure the quality of routing performance by path throughput represented as *PT*, which corresponds to the number of packets that can be sent for per unit time over a communication link as discussed in Refs. [1,3]. During Routing Update Period represented as $(RUP(\tau))$, a frequently used standard criterion for estimating the higher *PT* on different channels is the link-throughput denoted as $\psi_{ij}^{m}(\tau)$. Where $RUP(\tau)$ is defined as the time period between two route update packets being received as discussed in Ref. [22]. Consequently, the routing protocol can estimate the *PT* for every path P_i, i.e., PT_{p_i}, to maximize the network throughput. Therefore, *PT* can be estimated for any arbitrary path P_i during $RUP(\tau)$, by using the following equations:

$$PT_{p_i}(\tau) = arg\ max[P_1(\tau), P_2(\tau), \ldots P_i(\tau), \ldots P_I(\tau)] \tag{1}$$

where to reduce the probability of bottleneck for the path P_i that deteriorates the throughput for data-flow through P_i, we set P_i is as:

$$P_i(\tau) = arg\ min[\psi_{ij}(\tau), \psi_{jk}(\tau), \ldots, \psi_{fz}(\tau)] \tag{2}$$

where the "link-throughput", $\psi_{ij}(\tau)$, during $RUP(\tau)$ can be calculated by considering maximum link throughput from the set of available channels' throughput, as presented in the following equation:

$$\psi_{ij}(\tau) = arg\ max[\psi_{ij}^{1}(\tau), \psi_{ij}^{2}(\tau), \ldots, \psi_{ij}^{m}(\tau) \ldots, \psi_{ij}^{N}(\tau)] \tag{3}$$

where the parameter $\psi_{ij}^{m}(\tau)$ can be calculated as:

$$\psi_{ij}^{m}(\tau) = CAP_{ij}^{m}(\tau) \times CQ_{ij}^{m}(\tau) \times \psi^{m} \tag{4}$$

where $CAP_{ij}^{m}(\tau) \in [0,1]$ corresponds to the channel availability probability for channel *m* during $RUP(\tau)$ which is tantamount to the PU inactive probability p_{off}^{m} . For channel *m*, the parameter $CQ_{ij}^{m} \in [0,1]$ denotes the channel quality whilst ψ^{m} represents the idle link throughput [1].

For static CRNs, the position of each user remains constant and, therefore, the PU-CU relative distance is time-invariant. Therefore, it is possible to compute or estimate the values of CAP and CQ a priori or based on the channel occupancy and link quality history [29]. We refer to the name of "classical channel-aware method" on the methods to estimate $\tilde{\psi}_{ij}^{m}(\tau)$ for such a case [22]. Moreover, in static scenarios it is common practice to assume stationary network environment and data traffic. However, besides the time and link are invariant for both CAP and CQ such that $CAP_{ij}^{m}(\tau) = CAP^{m}$ and $CQ_{ij}^{m}(\tau) = CQ^{m}$. Moreover, the link throughput is governed by the PUs' activities over the different regional networks [13,29]. On the other side, for mobile scenarios, the PU-CU relative distances are expected to be time-varying [7,13]. It is therefore natural that in such a case, both the CAP and CQ will be both time- and link-variant. Differently from the "classical channel-aware method", in MCRNs, the link-throughput relies on the three central factors i) the Link-based $CAP^{m}(\tau)$, that is affected by the PU inactive probability $P_{off}^{m}(\tau)$ and the non-interfering probability of a pair of CUs (e.g., *ui* and *uj*) to the active PUs NA_{ij}^{m}; ii) Link-based $CQ_{ij}^{m}(\tau)$; and iii) idle channel throughput ψ^{m}.

To provide a better illustration, we consider an example in Fig. 1, where channels a, b, and c have throughput of 1 Mbps. Moreover, we set three active PUs on channels at time *t*. In the considered scenario, the CU_i wants to communicate with the CU_d during $RUP(\tau)$. For more details, we set three paths from the source CU_i to its destination node CU_d which are $P_1 = \{CU_i \rightarrow CU_j \rightarrow CU_h \rightarrow CU_d\}$, $P_2 = \{CU_i \rightarrow CU_k \rightarrow CU_r \rightarrow CU_d\}$, and $P_3 = \{CU_i \rightarrow CU_f \rightarrow CU_z \rightarrow CU_d\}$. In order to compute the $PT_{p_i}(\tau)$, we need to estimate every link_throughput, $\tilde{\psi}_{ij}^{m}(\tau)$ from the source CU_i to destination node CU_d, by applying the classical channel-aware method. We obtain $(PT_{p_i}(\tau))$ as the maximum bandwidth between all the paths from a source node CU_i to its destination CU_d during $RUP(\tau)$ as $PT_{p_i}(\tau) =$ arg max $[P_1(\tau), P_2(\tau), P_3(\tau)] = 0.35$ Mbps. Therefore, in context of the classical channel-aware method, the data passes through, is $PT_3(\tau) = CU_i \rightarrow CU_f \rightarrow CU_z \rightarrow CU_d$, and channel c represents the best channel. In contrast, with mobility scenarios, we need to compute every link-throughput from the source to its destination node during $RUP(\tau)$, and take into consideration the mobility situation of the CUs. According to this refined method, the maximum *PT* among all paths in mobile scenarios is $PT_{p_i}(\tau) =$ arg max $[P_1(\tau), P_2(\tau), P_3(\tau)] = 0.6$ Mbps. Therefore, the higher *PT* during $RUP(\tau)$, in the case of mobility, is $PT_2(\tau) = CU_i \rightarrow CU_k \rightarrow CU_r \rightarrow CU_d$, on channel b, respectively, not $PT_3(\tau)$, on channel c.

Due to a change in relative distance as a result of mobility, the link and channel are time-variant for CAP and CQ, which can affect the estimation of link-throughput. The higher throughput and a lower delay path will play a vital role to select the best path for transmitting data packets. The proposed SDRP will utilize a combined metric by considering such key routing features along with spectrum diversity for selecting the best path.

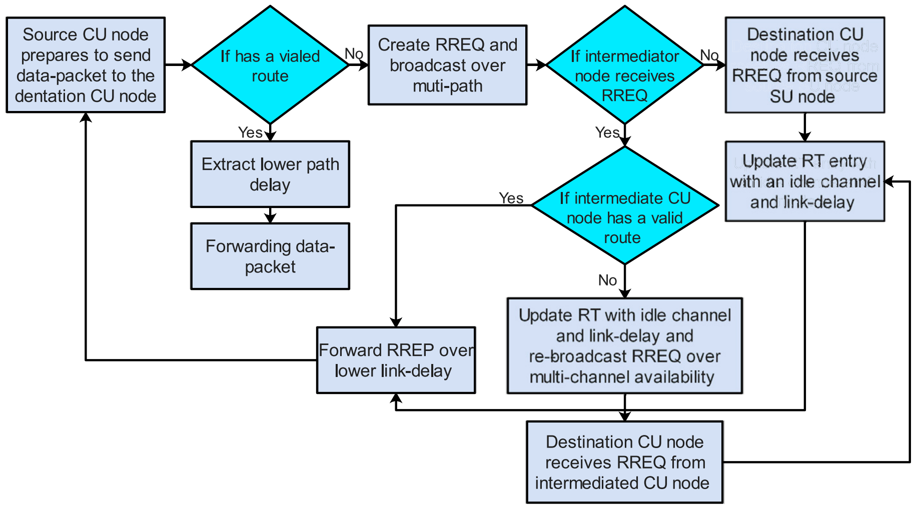

Fig. 3. SDRP workflow diagram for the cognitive user.

4. Overview of the proposed routing mechanism

In this section, a Software Defined Routing Protocol (SDRP) for Mobile Cognitive Radio Networks (MCRNs) by taking account of time-varying relative distances between PU and CU. It dynamically controls the data routing path using programmable computer instructions (e.g., software) and CRNs features to transmit data packets from an arbitrary source (denoted by *S*) to an arbitrary destination (denoted by *D*). An optimized SDRP for cognitive radio selects the best routing path for a given performance metric and exploits the cross-layering and flexible routing decision engine mechanism to counteract the mobility of PU and/or CU and dynamic spectrum access that will play an important role in CAP estimation. However, in the static case, it solely depends on PU activity, and prevents the adverse impact of PUs and CUs effects in MCRNs scenarios.

Spectrum-route selection in MCRNs is a very challenging task due to the useable spectrum variation and the assorted channel quality. Therefore, it is paramount to find practical solutions to empower CU to forward data-packet safely. The cross-layer SDRP focuses on developing and maintaining multi-hop paths between CU nodes by choosing a lower latency and higher throughput for the spectrum route. Therefore, an SDRP framework is proposed, including various system features to reach different layers for accurate information collection, analysis, and reaction. Concisely, this study follows a cross-layer design for a cognitive radio-routing protocol, which aims for reducing interference with PUs and optimize routing protocol capabilities in MCRNs. On the other side, the main challenge of implementing the proposed idea with the existing protocols on CRNs is how to control the routing overhead due to the dynamic CRNs topology, which results in more control packets to the updated routing table for the next hop. In the same context, the proposed protocol needs to trade-off the routing update period time and channel estimation time for successful routing performance.

Functionalities of the proposed SDRP are explained in details in the following subsections.

4.1. Cross-layer framework for SDRP

In order to exploit the channel behavior, it requires participation from other layers (e.g., physical, link layer, etc.) with the network layer [30]. Nevertheless, the layers in the network stack model have not supported to fast access this information between layers [31]. Usually, this control information is exchanged between adjacent layers through the Service Access Point (SAP) that supply access to selected protocol operations through accurately specified primitive activities based on the Open Systems Interconnection (OSI) model. In other words, the standard OSI model can not meet the requirements of the routing algorithm in MCRNs environment. The cross-layer design is an indispensable solution to tackle the routing algorithm's challenges in the MCRNs environment. Furthermore, SDRP has introduced a new mechanism to joint routing and cross-layer information. It offers a suitable opportunity to provide the SDRP with the necessary information about idle channels to overcome the phenomenon of time-variance regarding channel availability and routing adaption. Fig. 2 describes a new framework that will assist in designing a cross-layer routing protocol for MCRNs. When a CR user *S* needs to send its data-packets to destination user *D*, *S* received a request from the application layer. The data will be sent to the destination by exploiting periodical updated routing information. If the routing path is not available or not updated, the source will explore the suitable path using SDRP route discovery mechanism. The details of the SDRP are as follows.

4.1.1. Channel estimation and channel access mechanism

Time-variant channel estimation is considered a crucial technique to develop and improve the performance to detect an idle license channel in MCRNs. It includes more than one factor, such as CAP, CQ, and link-interference, and each of these factors is regarded as an essential key to narrow the gap of channel selection challenge under the time-variant relative distance. Moreover, this information is exchanged with the MAC layer to schedule and select the idle license channels according to the channel access mechanism. The MAC layer will also come to play through sharing its data and services with the network layer to enable a routing method to access and choose the available channel according to the utilizing dynamic routing metric. In contrast, the estimation unit has trained to update the situation of license channels every RUP (τ) to counteract the time-variant effect on PU and CU. Consequently, the channel information will be updated periodically. Thus, the cross-layer method assists the channel information that is obtained from other layers to join the process of the routing layer.

4.1.2. Cross-layer mechanism

It acts like a transit point where the cross-layers information is gathered. It performs the proposed cross-layer superposition cooperation mechanism at the network layer. In fact, it operates as the unit of accumulation and traffic scheduler. Furthermore, Fig. 2 also reveals that the cross-layer mechanism feeds up the other units with the necessary information to perform the appropriate operations.

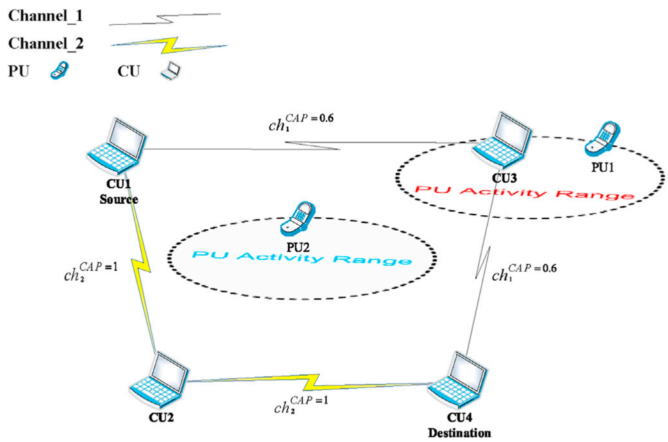

Fig. 4. The network setup for performance evaluation.

4.1.3. Node and time variant (NTV) decision engine

It is a new feature with numerous functions and connections with other layers, which helps in understanding the layer stack's information through a cross-layer mechanism to make the appropriate routing decisions. In practical terms, it is accountable for enhancing and managing routing decisions based on some input parameters from other layers and combine it with the decision engine. Fig. 2 illustrates that the NTVDE can estimate several features such as link-throughput, path-throughput and link-delay. Therefore, SDRP enables high flexibility and intelligence to make a correct decision about the channel and route selection for MCRNs. The feedback from NTVDE will enable the routing algorithm in MCRNs to select a suitable path (i.e., with higher throughput and lower delay) out of the licensed frequency, that is arranged in N mutually exclusive bands/channels. For that, the CU user needs firstly to estimate the link-throughput $\tilde{\psi}^m_{ij}(\tau)$, which is mainly depended on the CAP, and CQ over a channel $m \in \{1, \ldots, N\}$ as proposed in our previous work [1].

Moreover, estimation of the channel availability is needed during the sensing time slot (T_k), denoted as $\tilde{p}_{ij,m,\mathcal{L}}(T_k)$, for a channel m in order to estimate CAP during the next RUP (τ) as presented in Equation (5) [1].

$$\tilde{p}_{ij,m,\mathcal{L}}(T_k) = \begin{cases} 1, & \text{if } \left(\tilde{d}_{i,l}(t) > R_{i,l}\right) \text{AND} \\ & \left(\tilde{d}_{j,l}(t) > R_{j,l}\right), \forall l \in \mathcal{L}, \\ \prod_{l \in \mathcal{L}} P^{off}_{l,m}, & \text{otherwise,} \end{cases} \tag{5}$$

For all $t \in T_k$, where $R_{i,l}$ and $R_{j,l}$ denotes the interference ranges of the channel m that belongs to a set of $\mathcal{L}$ licensed users, $\tilde{d}_{i,l}(t)$ and $\tilde{d}_{j,l}(t)$ denotes the approximated distances between the i-th CU and l-th PU, and the j-th CU and l-th PU at time $t \in T_k$, accordingly.

The channel availability probability for the next RUP can be estimated, $\tilde{CAP}(\tau)$, using Equation (6) which is presented as:

$$\tilde{CAP}^m_{ij}(\tau) = \frac{1}{q}\sum_{k=0}^{q-1} \tilde{p}_{ij,m,\mathcal{L}}(T_k) \tag{6}$$

where $\tilde{p}_{ij,m,\mathcal{L}}(T_k)$ is defined in Equation (5). The estimated channel availability $\tilde{CAP}^m_{ij}$ can be calculated from averaging of $\tilde{p}_{ij,m,\mathcal{L}}(T_k)$. Moreover, the channel quality in MCRNs varies over time due to path loss and fading as discussed in Ref. [32]. As a result, The CU with more reliable channel quality will receive higher throughput. Therefore, the channel quality factor can be integrated to detect the best idle channel to improve the channel selection mechanism. The channel quality can be estimated by Equation (7) as presented in Ref. [1].

$$\tilde{CQ}^m_{ij}(\tau) = \frac{T - T_{\text{sen}}}{T \times W \times \log_2|\mathcal{X}|} \cdot \frac{1}{q}\sum_{k=0}^{q-1} R_{i,j}\left(\tilde{\text{SNR}}_{m;k}, \tilde{g}_{m;k}\right) \tag{7}$$

where T and T_{sen} represent the single frame and sensing times, the bandwidth of channel m is denoted by W, $\mathcal{X}$ is the constellation set coming from the modulation, and q is the number of frames per location update interval. The value of $R_{i,j}(\tilde{\text{SNR}}_k, \tilde{g}_k)$ is the calculated rate per frame which is estimated by $\tilde{\text{SNR}}_{m;k}$ and the actual fading $\tilde{g}_{m;k} = [\tilde{g}_{m;k,1}, \ldots, \tilde{g}_{m;k,B}]$, where all these equations has been already studied in our previous work [1]. Finally the Link Throughput between CUs i and j for a channel m can be estimated during the RUP(τ) and presented as:

$$\tilde{\psi}^m_{ij}(\tau) = \tilde{CAP}^m_{ij}(\tau) \times \tilde{CQ}^m_{ij}(\tau) \times \psi^m \tag{8}$$

SDRP will exploit $\tilde{\psi}^m_{ij}(\tau)$, in order to estimate the dynamic routing metric, which is discussed in the next subsection.

4.1.4. Proposed metrics for SDRP

In this subsection, we will discuss two metrics that will be utilized to select the best medium of a link for a pair of CUs and best path selection for a pair of source and destination. In both cases, the selected medium of a link and path will ensure higher throughput and lower delay.

Metric for link: To estimate the best medium for a link, it is required to estimate the link's delay according to the available channels. As the channels' capacity fluctuates in a real world environment, the link's delay will be different according to the considered channel. For this purpose, we first present the following proposition.

Proposition 1. *The expected link delay $\mathcal{D}^m_{l_{ij}}(\tau)$ of a packet sent by an arbitrary CU, u_i, to the neighbor CU, u_j, according to a channel m for the next RUP τ is given by*:

$$\mathcal{D}^m_{l_{ij}}(\tau) = \frac{1}{\tilde{CAP}^m_{ij}(\tau) \times \tilde{CQ}^m_{ij}(\tau)}\left(\frac{L}{\psi^m_{ij}(\tau)}\right) \tag{9}$$

where $\tilde{CAP}^m_{ij}(\tau)$ is defined in Equation 6, $\tilde{CQ}^m_{ij}(\tau)$ is defined in Equation 7, L is the packet length and $\psi^m_{ij}(\tau)$ is the idle link-throughput.

Proof. It can be easily estimated that the delay of a link depends on the packet size and throughput of the channel m if and only if the m is available. However, in the MCRNs, the availability of m depend on the $\tilde{CAP}^m_{ij}(\tau)$ and $\tilde{CQ}^m_{ij}(\tau)$. For instance, if the $\tilde{CAP}^m_{ij}(\tau) = 1$ and $\tilde{CQ}^m_{ij}(\tau) = 1$ (i.e., available channel for entire RUP τ), the delay can be approximated using parameters L and $\tilde{\psi}^m_{ij}(\tau)$; otherwise, the additional delay may be incurred due to the CAP and CQ. The expression (9) *can be used to calculate the minimum expected delay, as presented in* [29].

Remark 1. The expected link delay according to the channel m provides us the ability to approximate the packet delay over a link, considering:

i) the packet transmission delay, inversely proportional to the channel capacity $\tilde{\psi}^m_{ij}(\tau)$
ii) the delay coming from the PU activity, i.e., $\tilde{CAP}^m_{ij}(\tau)$.

We estimated the delay for every RUP τ. During the entire RUP, the same value is utilized for selecting the channel and updated the value after finishing the RUP. Also, this metric participates in mitigating the effect of propagation delay.

From Proposition 1, we can now derive the expression of metric for a link.

Proposition 2. *The expected link delay $\mathcal{D}^{Best}_{l_{ij}}(\tau)$ of a packet transmitted by an arbitrary CU, u_i, to the neighbor CU, u_j, for the next RUP τ is given by*:

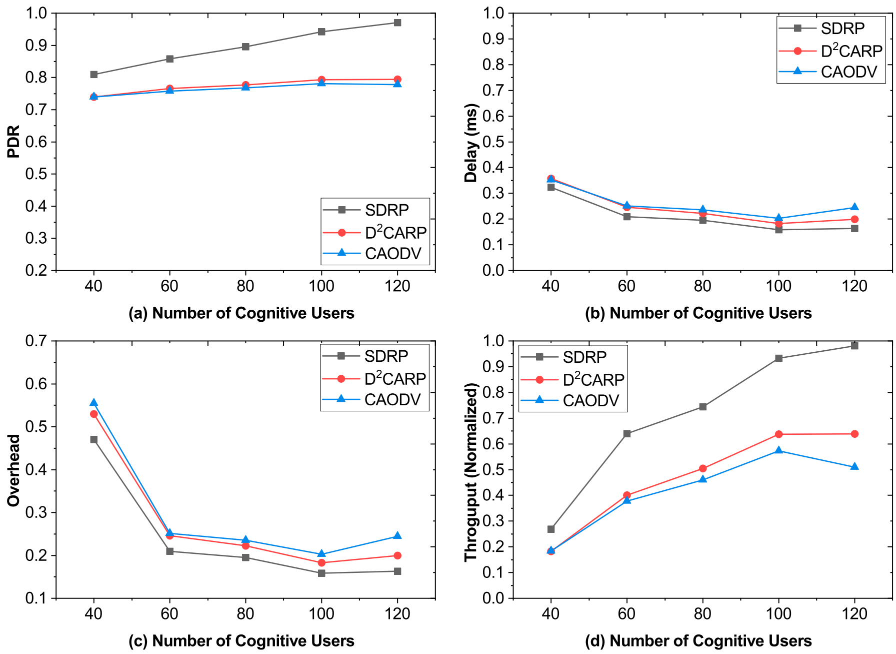

Fig. 5. Various performance metrics against the number of cognitive users for Scenario I.

$$\mathcal{D}_{l_{ij}}^{Best}(\tau) = \min(\mathcal{D}_{l_{ij}}^{1}(\tau), \mathcal{D}_{l_{ij}}^{2}(\tau), \dots, \mathcal{D}_{l_{ij}}^{N}(\tau)) \tag{10}$$

where the available channel set is {1,2,..N}. The delay according to the channel can be estimated using Proposition 1.

Proof. *Proposition 2 can be directly proof by using Proposition 1.* □

Remark 2. . Each intermediate node will forward the data packet through a channel with minimum delay. If there is more than one channel with similar delay, the channel will be randomly selected to transmit data packet toward the neighbors. Expected Transmission Count (ETX) is a standard selection method for measuring delay in wireless ad hoc networks. Employing probe packets for its estimation presents significant overhead in the context of multi-channel routing in CRNs since every point has to perform the same task over multiple channels. This method also oversimplifies both the channel availability and quality in a single numerical quantity. In contrast, our metric tailored for MCRNs, that uses the CAP to incorporate the time-variation effect on link quality during the routing update period.

Metric for Path: To estimate the best path between the source and destination, we need to estimate the delay of a path among the available paths. As SDRP discovers a multi-path during its route discovery phase, channels' capacity and the number of hops varies among the available paths. As a result, the delay of the paths will be different. For this purpose, we introduce the following proposition.

Proposition 3. *The expected path delay $\mathcal{D}_{p_{s,d}}^{x}(\tau)$ experienced by a packet sent by an arbitrary source CU s, to the destination CU d, according to a node disjoint path x during the RUP τ is given by*:

$$\mathcal{D}_{p_{s,d}}^{x}(\tau) = \sum_{h=1}^{H} \mathcal{D}_{l_h}^{Best}(\tau) + NPD_h \tag{11}$$

where $\mathcal{D}_{l_h}^{Best}(\tau)$ is defined in Equation 10, h represent the hop number of the routing path and NPD_h is the node processing delay.

Proof. The delay of a path can be easily estimated by adding the minimal delay of each link of the path along with the node processing delay.

Remark 3. The expected path delay enables estimation of packet delay over a path using:

i) the best delay that was estimated for a link (i, j) at hop number h, and
ii) the delay due to the received packet processing time.

We estimate the path delay of a pair of source and denotation (e.g., (s, d)) for every RUP τ. During the entire RUP, the same value will be utilized for selecting the path for such (s, d) and the resultant values will be updated.

By using Proposition 3, we can now derive the expression of metric for a path.

Proposition 4. *The expected best path delay $\mathcal{D}_{p_{s,d}}^{Best}(\tau)$ experienced by a packet sent by an arbitrary source CU s, to the destination CU d, during the RUP τ is given by*:

$$\mathcal{D}_{p_{s,d}}^{Best}(\tau) = \min(\mathcal{D}_{p_{s,d}}^{1}(\tau), \mathcal{D}_{p_{s,d}}^{2}(\tau), \dots, \mathcal{D}_{p_{s,d}}^{x}(\tau), \dots, \mathcal{D}_{p_{s,d}}^{Z}(\tau)) \tag{12}$$

Where the available path set is {1,2,..Z}. *The delay according to the path can be estimated using Proposition 3.*

Proof. *Proposition 4 can be directly proof by using Proposition 3.*

Remark 4. A Source CU will forward the data packet through a path which offers minimum delay. If there is more than one path with similar delay, it will be randomly selected to transmit data packet towards the destination. On the other side, ETX selects a path according to the sum of the probing results in each link in the path.

4.1.5. Routing operation phase

The communication demand at the CU source triggers the SDRP operation phase. Fig. 2 discovers that the routing operation phase has included different components. In contrast, the NTV decision engine cooperates with its services within the routine operation phase. For instance, selecting the best link for a pair of CU users at the NTV decision engine can contribute positively to the packet forwarding at the routing operation phase. Also, the unit is connected bidirectional with the routing table. It is reasonable because the routing table needs to share its information as the routing metric with the routing phase to choose the best path between source and destination. Respectively, the routing operation phase has to announce the routing table about any updates such as link-break, PU-error, etc.

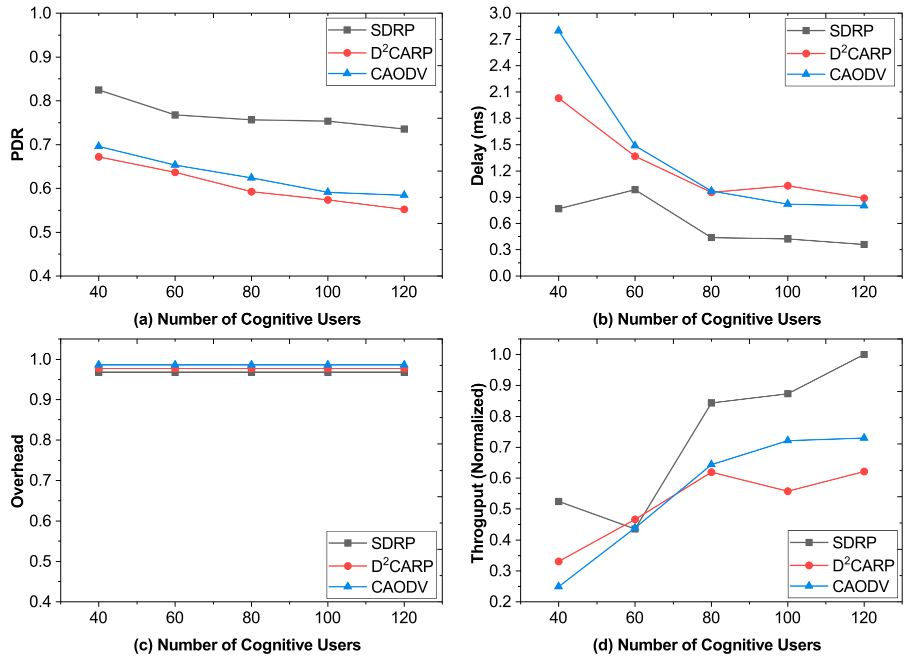

Fig. 6. Various performance metrics against the number of cognitive users for Scenario II.

4.1.6. User of SDRP

Routing protocol operations for CUs in CRNs are not similar to the ad hoc network protocol. The CUs need to discover the temporary free spectrum over multi-channel to establish a stable route. To assist CUs in setting up the right routing path over the best ideal channel, the SDRP has incorporated new functions into the routing operation phases. As indicated in Fig. 2, the CU can act as a source, intermediate, or destination. On the other side, CUs and SDRP operations have a mutual interest in exchanging and updating the routing operations knowledge to send, forward, or receive data packets safely by CUs. For more details, the following sections will discuss the SDRP operations.

4.2. Routing operation for user of SDRP

SDRP belongs to a reactive distance-vector routing protocol designed using the cross-layer method for operating in MCRNs and inherits some D^2CARP and CAODV features [18] such as joint spectrum and path diversity, and primary user route error (PU-RERR). Unlike D^2CARP and CAODV, the channel probability model, inside SDRP, computes the CAP for every channel list every $RUP(\tau)$. In contrast, the D^2CARP and CAODV select an idle channel based on PU activity and off probability function. If so, then the channel is available (i.e., equal to one). Otherwise, the channel is unavailable (i.e., equal to zero). In that case, the D^2CARP and CAODV select the idle channel randomly without considering any criterion, which is not considered the perfect method for selecting the channel-route for MCRNs routing protocol [18]. The proposed algorithm's complexity is similar to the existing D^2CARP, as we follow the same workflow. However, our efficient path selection technique improves the performance of the protocol.

SDRP workflow and the rules which CU has to follow during transmission of data packets are shown in Fig. 3. The workflow diagram shows that when the CU intends to transmit the data packet, it first needs to examine the routing table to extract a valid lower-delay path. If so, the CU then starts to send the data packet to a target CU node. Otherwise, SDRP manages the process of route discovery through an announcement by CU for generating RREQ over multiple idle channels. In the meantime, an intermediate CU receives that RREQ and it will forward Route REPly RREP over a lower delay link. In contrast, if the intermediate CU has not had a valid route, then it will update its routing table entries and rebroadcast the RREQ signal. On the other side, the destination CU node is possible to receive RREQ directly from the source/intermediate node according to its position. Either way, the destination CU node will update its RT information and create RREP for direct forwarding to the source CU node or over an intermediate CU node. Then, the source will compute the multi-path delay for forwarding data-packet over the lowest-delay path.

Remark 5. In order to establish the path, additional signaling information needs to be sent, which is considered as an overhead. If the path is not robust, then the source node needs to CALL Route Discovery phases frequently, which increases the overhead. However, as the proposed path selection is more robust than the existing techniques, since it considers not only the CAP but also CQ, a less number of time Route Discovery phases need to be called. Therefore, the overhead of the proposed method is less than those of the existing techniques.

As SDRP selects the best path by using cross-layer acknowledgment during forwarding data packet, it maximizes spectrum productivity, reduces channel switching, and interference with PU.

4.2.1. SDRP main function

Algorithm 1 presents functions to manage the SDRP operations. When an attribute CU node wants to send a data packet to a target CU node and has a valid route, it will then call the forward data-packet function from (steps 3–9). Otherwise, it starts the process of the route discovery phase through an RREQ function from (steps 12–18) which illustrates that the SDRP handles the routing control packet according to the nature of the routing operations. In more, the SDRP will call a specific function according to the type of a control packet related to RERR, RREQ, or RREP. The software-defined concept is expected to benefit from context-awareness, whereas the routing engine's dynamic is dependent on the availability of cross-layer information sources in order to operate efficiently and optimize the routing algorithm by making the proper routing decision. The following subsection will further discuss new features of the SDRP function for CUs.

Algorithm 1 Software-defined Routing Protocol

```
1: function Main(Routing Operations)
2:     //Initialization
3:     if an arbitrary node has to handle Data Packet then
4:         //Handling Data Packet
5:         if source node has valid route to destination node then
6:             Forwarding − data − packets(source, destination)
7:         else
8:             RREQ(source, destination)
9:         end if
10:    else
11:        //Handling Control Packet
12:        if it is a RERR then
13:            RERR(source, destination)
14:        else if it is a RREQ then
15:            RREQ(source, destination)
16:        else
17:            RREP(source, destination)
18:        end if
19:    end if
20: end function

21: function Forwarding-data-packets(source,destination)
22:    if it is a source node then
23:        -Check RT entry
24:        -Extract the available paths for the source and destination pair
25:        -Choose the best path according to the path delay
26:        -Forward Data Packet to next hope according to the best path
27:    else if it is an intermediate node then
28:        -Choose the best link according to the link delay for this particular source and destination pair
29:        -Forward data packet to Next hope according to the best Link
30:    else
31:        -A Packet has been received and acknowledgment will be sent to source
32:    end if
33: end function

34: function RREQ(source,destination)
35:    // Intermediate CU node handles with RREQ Control Packet that generates by Source CU node
36:    if RREQ is not recived before then
37:        if RT entry has a valid route then
38:            -Send RREP over lower link − delay
39:        else
40:            -Estimate link − delay for pair nodes
41:            -Sensing channel availability
42:            -Update RT entry by set a reverse channel-link route related to channel i and link − delay
43:            -Update control RREQ packet header and re-broadcast over idle channel(s)
44:        end if
45:    else if it is additional RREQ recives but on different channel-link then
46:        -Update RT entry by set a new reverse route through that channel-link
47:        -Update RT link − delay estimation entry
48:        -Update RT Channel availability entry
49:    else
50:        if it is the new or better RREQ recived then
51:            // Better RREQ according to Link − delay
52:            -Update RT entry by adding a new reverse route include a new channel i and lower link − delay
53:        end if
54:    end if
55: end function

56: function RREP(source,destination)
57:    if a node is a destination then
58:        // Destination CU node revives RREQ and generates RREP
59:        -Update RT entry related to link − delay and channel availability
60:        -Create a forward route and forwarding control RREP packet over lower link − delay for the next CU node
61:    end if
62:    if a destination node recived more RREQ over a new idle channel/path then
63:        -The destination node send additional RREP
64:    end if
65:    if an intermediate CU node recives RREP then
66:        -Update RT entry related to link − delay and channel availability
67:        -Update RREP header and forward it over lower link − delay
68:    else if it is extra RREP from neighbor node but over different channe-link then
69:        -Update RT entry related to link − delay and channel availability
70:        -Build a forward route and forward RREP through that channel
71:    else
72:        if source CU node recives first RREP then
73:            -Compute forward path−delay between source CU node and destination
74:            -Update forward RT entry and channel availability
75:            -Forwarding data-packet over lower link−delay for pair nodes
76:        else if it is extra RREP ove new channel-link then
77:            -Update RT entry with new idle channel(s)
78:            -Update estimation link−delay between source and next node
79:        else
80:            if it is the new or better RREP over a new path then
81:                // According to the Path-Delay metric
82:                -Compute a new forward path−delay between source CU node and destination
83:                -Update RT with a new forward route through the channel i
84:                -Select a new lower path − delay and forwardness data-packet
85:            end if
86:        end if
87:    end if
88: end function

89: function RERR(source,destination)
90:    if RERR due to PU then
91:        -The link is temporarily suspended due to the PU arrival
92:        -Inform the Source through the Neighbor's Node
93:    else
94:        -The link breakage due to node mobility or poor channel quality
95:        -Inform the Source through the Neighbor's Node
96:    end if
97: end function
```

4.2.2. *SDRP for source cognitive user*

At the time of sending the data packet, the source CU examines its routing table entry. If it has a valid route to destination CU, it selects the valid route with higher CAP and lower path delay, as it is shown from (steps 22–26). If there is no valid route, then source CU initiates a route discovery phase by calling the RREQ function and broadcast the control RREQ packet to all CU neighbors, as shown from (steps 34–55).

On the other hand, during the route discovery phase, the source CU calls the RREP function when it receives the first RREP packet (steps 72–75). Step 73 illustrates that the source CU node computes the path delay for the destination CU node. The source CU node's routing table entry is updated related to channel availability to select the free higher idle channel (step 74). Then source CU node calls the forwarding function to send a packet across the lower link delay (step 75). Because of the joint path and spectrum diversity nature, the source CU may receive more than one RREP over various idle channels (step 76). Therefore, the source CU updates a forward route through a new idle channel link for the lower path delay (steps 77–78). This future enables source CU to create a forward route through different unused channels. Moreover, due to multipath features, the source CU node might receive a new/better RREP (step 80). The source CU node updates its routing table entry by creating a new forward route and selects the lower path-delay for starting to send data-packet over it (steps 82–84).

The phenomenon of PU-RERR occurred because of PU activity and CU mobility. The function of RERR generates the PU-RERR control packet to announce the source CU node through neighbor nodes that channel-link is not valid now (steps 90–92). One of the benefits of selecting the highest channel availability, according to cross-layer information, is reducing the rate of channel switching, which takes part in reducing routing overhead. On the other hand, a Route Error (RERR) is fed back to the source for a broken link (steps 94–95). Therefore, the source CU will invalidate the route and re-perform the route discovery if needed.

4.2.3. *SDRP for intermediate CUs*

Ultimately, CU intermediate node receives a packet from the source or its neighbors, where the packet is either a control packet or a data packet. CUs must sense the channel before receiving or transmitting over it and avoid any interference with PU activity. Intermediate CU recognizes the received packet to identify its type (i,e., data-packet or control packet) and selects the lower link delay for forwarding a data packet to the next CU node, as shown from (steps 27–32). The intermediate CU node receives the RREQ (perhaps the destination CU itself) with a current route to the destination. When the intermediate CU receives the RREQ, it will first check if the RREQ has not been received previously from (step 36). If so, then the intermediate CU checks if it has a valid path for the destination CU or not from (step 37). Otherwise, intermediate CU will estimate the link delay for a pair of nodes and then attach this information to its routing table by creating a reverse path, incorporating the free channel availability received through that RREQ (steps 39–41). Afterward, SDRP updates the control RREQ header and re-broadcasts that RREQ over all unoccupied channels by PU from(step 42).

Intermediate CU is exposed to receive more RREQ for the same source and destination, but on a different channel-link from (steps 45–48). In more detail, the intermediate CU will create a new reverse path through the channel$i$, which receives the RREQ over it. Then, it updates the value of link delay and channel availability related to the new reverse path. When an intermediate node receives a new or better RREQ related to lower link delay, the reverse route is updated by including a new channel, say i, and lower link delay from (steps 50–53).

Once an intermediate CU obtains the first RREP, it then constructs a forward route, say over channel i, and updates RT relating to link-delay and forwards RREP from all channels that are available in the reverse path table (steps 62–63). Meanwhile, if an intermediate CU receives further RREP over a new channel frequency, it then updates RT entry related to link delay and channel availability (steps 64). Then, a forward route is constructed), which then pushes RREP solely over that channel

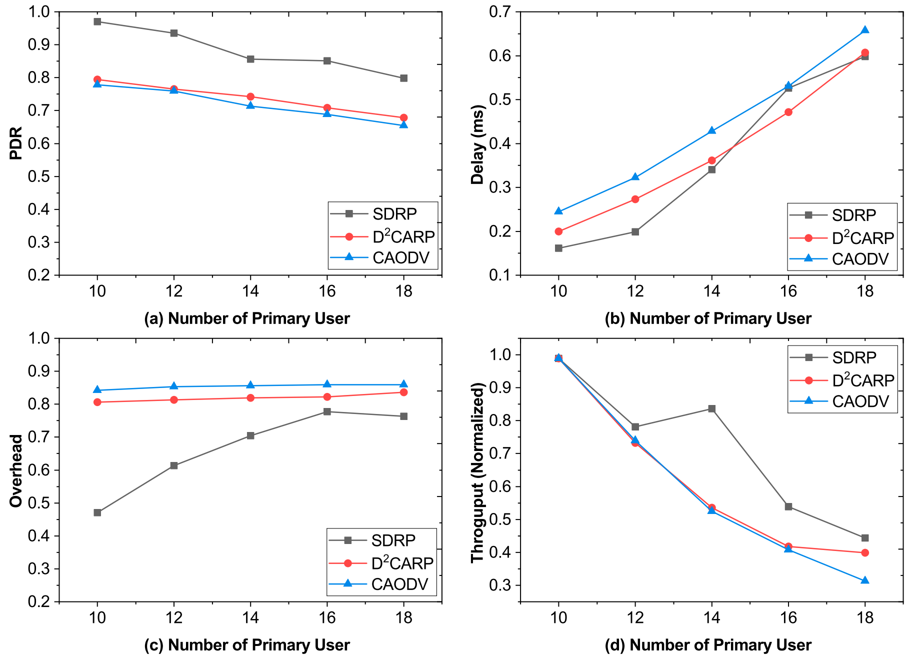

Fig. 7. Various performance metrics against the number of primary users for Scenario III.

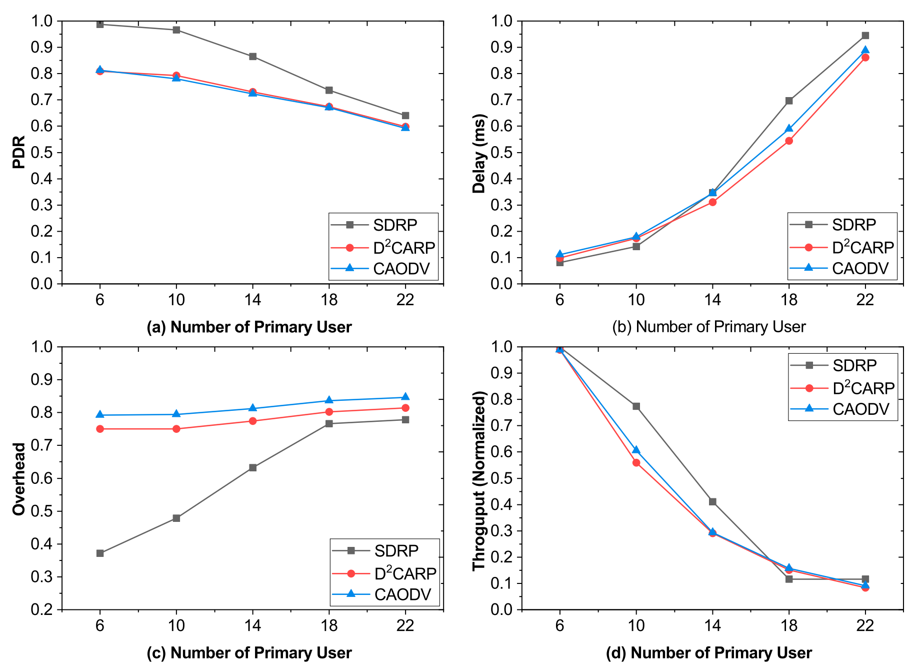

Fig. 8. Various performance metrics against the number of primary users in a large network for Scenario IV.

(step 65–67). From this perspective, the intermediate CU node can generate a list of multiple idle channels, based on cross-layer information. Moreover, when CU intermediate gets a new RREP, it updates its forwarding-path table over the channel.

4.2.4. SDRP for destination cognitive user

Eventually, the destination CU node will receive a control or data packet over different free channels and paths while detecting if the channel is idle and identifying the packet type. In the case of the data packet, the destination CU just received that packet with no further action. Moreover, when the destination CU obtains the first RREQ, it sets a backward path. Afterward, it records the CU neighbors' addresses that it received with the first RREQ over an idle channel as soon as the first RREQ has been received, as shown in (steps 57–61). In detail, the destination node updates RT entry related to link delay and channel availability, it then creates a forward route (step 59). The destination node setups a forward route to forward control RREP over lower link delay (step 60). On the other hand, due to the spectrum's diversity, the destination CU node is an exhibition to receive more RREQ over a new idle channel or path. As a result, the destination node will send additional RREP over different channels, as shown from (steps 62–64).

Remark 6. Software-defined Networking (SDN) is a new network architecture that separates the control panel and data plane to improve the

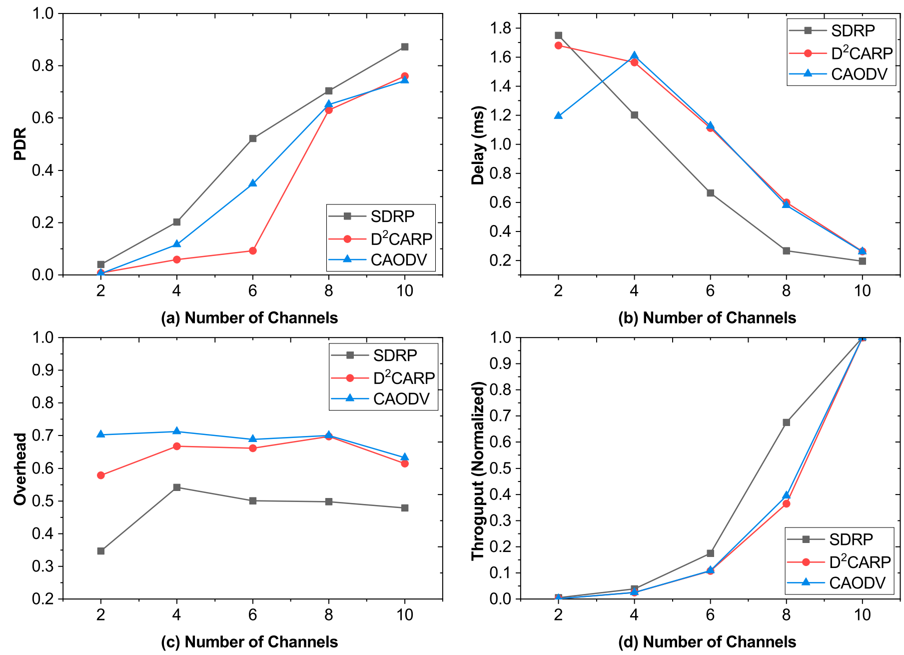

Fig. 9. Various performance metrics against the number of channels for Scenario V.

network performance [33]. On the other hand, the SDRP implements the cross-layer approach to collect the stack layers' information / services and share them with the routing engine (i.e., control plane). Meanwhile, the CU receives the decision of the routing engine for forwarding (i.e., data plane) the data packet to the destination.

5. Performance evaluation

In this section the SDRP performance is validated by carrying out a comparative study with widely-known D^{2}CARP and CDOAV protocols using the network simulator (ns-2). CAODV is designed to explore path and spectrum diversity in CRNs, whereas that D^{2}CARP considers joint path and spectrum. Consequently, D^{2}CARP and CAODV are considered as a reference in our work. For a fair comparison, we set the same simulation setup adopted in Refs. [18,19]. To figure out the influence CAP time-varying on routing performance, we only included the impact of CAP in MCRNs rather than CQ during path and link selection. In order to evaluate the performance, different metrics are used to compare and compute the contrasting behaviors of these various protocols, namely, Packet Delivery Ratio (PDR), overhead, delay, throughput, and PUs activity.

5.1. Network model

A. CU model: CU changes its trajectory based on a popular Random Way Point Mobility (RWPM) model [14] around a square area $\mathcal{A}$. The network model is considered as depicted in Fig. 4. An example can be given here where a source CU, namely CU_1, is within a range of a destination CU, namely CU_4, over idle license channels. At a specific time, channel 1, experiencing the PU activity, has CAP = 0.6, while channel 2, free from the PU activity, provides CAP = 1. During the time of using the licensed spectrum by CU, CU_1, has to sense channel before transmission on it, in order to circumvent interfering with PU. For this purpose, the activity of using license spectrum by the CU is regularized into fixed-sized intervals of duration T each. Furthermore, each slot T is divided into an interval of spectrum sensing T_s and a transmission interval T_{tx} to transmit CU data packets.
B. PU Activity Model: PUs in consecutive slots are subdivided into periods of OFF and ON which is followed by two independent exponential distributions with death rate $\alpha_{l,m}$ and birth rate $\beta_{l,m}$ [1,13,22]. These variables can be predicated using off-line statistics, local sensing information, or over tacit feedback in full-duplex communications [3]. The ON probability for the PU over channel m can be estimated by applying Equation (13) [1,13]:

$$P^{on}_{l,m} = \frac{\beta_{l,m}}{(\alpha_{l,m} + \beta_{l,m})} \tag{13}$$

Similarly, the OFF probability for the PU over channel m can be estimated by applying Equation (14) [1,13]:

$$P^{off}_{l,m} = 1 - P^{on}_{l,m} \tag{14}$$

Furthermore, we implemented two PU spectrum models, namely the Single PU for Channel (SPC) and Multiple PUs for Channel (MPC) [13].

5.2. Simulation setup

CUs travel in a square region based on the Random Waypoint model, where the node density has been set equal to 400 nodes/km^2. Each CU has coverage (communication range) of 120 m following the specification in the IEEE 802.11b standard and assumes the Two-Ray Ground propagation model. CU's speed movement is equal to 2 m/s, and routing update period (τ) is set at 5s. On the other hand, the PUs are immobile, and the transmission range is set at 300 m. PUs' operations are modeled on/off according to a two-stage process. The data traffic model considers Constant Bit Rate (CBR) data packets of size 1000 bytes. These data are transmitted over User Datagram Protocol/Transmission Control Protocol (UDP/TCP) connections with each user, creating a single data flow to a randomly chosen destination. Each run period is 1060 s with the active period for data traffic within [60s, 1000s]. Each experiment is performed 5 times, followed by calculation of the average values and the standard deviations for each metric. The routing metrics used to evaluate simulation results of the SDRP routing protocol's performance and compare it to other protocols are summarized below.

- Packet Delivery Ratio (PDR): It is the ratio of the source's packets sent to the destination's packets received. This metric indicates how well a routing protocol delivers data packets from source to destination. The

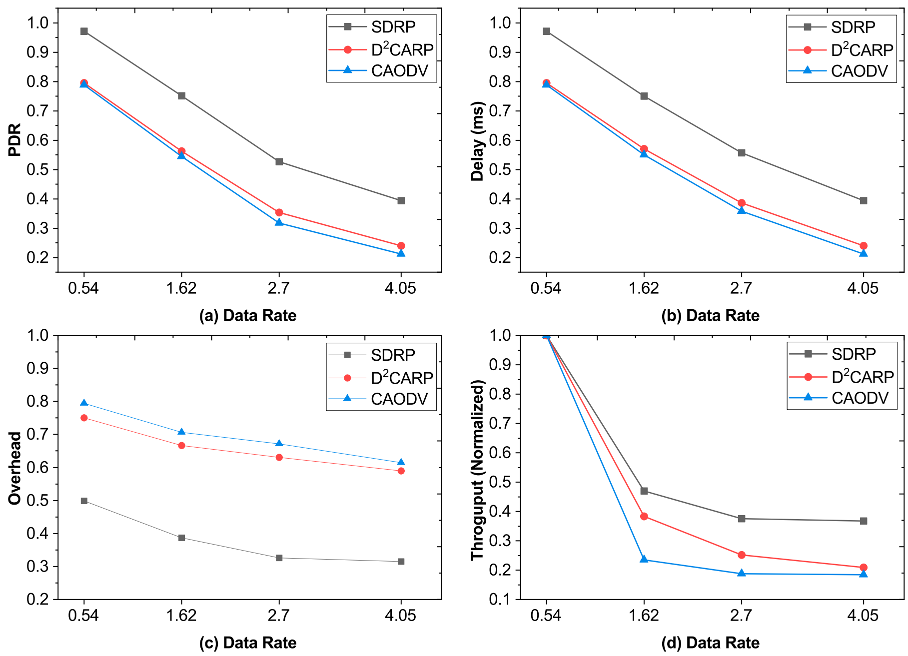

Fig. 10. Various performance metrics against the target data rate for Scenario VI.

higher the ratio, the better the routing protocol's efficiency. This metric may also affect network throughput.

- Average end-to-end delay: The average time it takes for all packets to travel from the source to the destination. This is determined by averaging the delays of all packets sent from all sources over the simulation period (i.e., the sum of time spent delivering packets to the destination).
- Routing Overhead Ratio: The ratio of the number of control packets sent for route setup to the number of active data packets received is known as routing overhead (or control overhead). This measure represents the cost of using a routing protocol.
- Throughput: It is the average rate of effective packet transmission per second from source to destination. As a result, throughput determines how efficient a routing protocol is at receiving data packets by destination.

5.3. Numerical results

In this section, numerous scenarios has been considered to explore the effect of the number of CU nodes, PUs, specific data-rate and channel-number. The various settings have been set in each scenario and discussed as follows.

5.3.1. Performance of SDRP with respect to CU diversity

The performance behavior of SDRP is compared and analyzed along with D^2CARP and CAODV by increasing the number of CUs. The number of PUs are set to 10 with the PUs' activities time given by 200s. CBR traffic is generated over UDP with data rate = 0.54 Mbps. It is observed from Fig. 5(a) that the SDRP achieves significant improvement in terms of PDR compared to D^2CARP and CAODV. It is also observed that when the CU (CU = 40 or CU = 60) is low, the PDR is low. The reason for this is that some intermediate CUs along the path may be located within the PU activity region, making them extremely sensitive to PU interference, resulting in a reduced PDF rate. In contrast, the PDR performance achieves a higher value up to approximately 90% of delivered packets for a higher number of CUs. The increasing number of CUs with fixed PUs can help CUs to stay out of the interference range of PUs. In both cases, with lower and higher number of CUs, the SDRP can get the highest PDR performance comparing with D^2CARP and CAODV. This can be justified because selecting the channel with the higher CAP has a significant impact on the route's stability, which increases the number of data transmissions and, as a result, lowers the data-packet drop rate.

Fig. 5(b) presents the performance comparison of SDRP with D^2CARP and CAODV using delay as a metric. SDRP can achieve a lower delay compared to D^2CARP and CAODV with a small number of CUs. Furthermore, the average delay fluctuates with the diversity the density of CUs is due to the change of a number of CUs and time-variant for channel availability. The light density of CUs and switching from one frequency domain band to another can restrict access the data packets to the CU destination, increasing routing delay. On the other hand, with a higher CUs density, more CUs will be paid out of the PU transmission range, reducing routing delay. However, increasing the number of CU demands more free spectrum, which is not always available due to PU activity. In the end, the SDRP can win a lower delay than D^2CARP and CAODV because of the robustness in the selection of high channel stability and a lower probability of interference with PU.

It is observed in Fig. 5(c) that the SDRP can achieve lower overhead as compared to D^2CARP and CAODV with a small number of CUs. Besides, when the number of CU increases, the SDRP can significantly outperforms as compared to D^2CARP and CAODV. This behaviour can be described by the fact that the CU can handle PU activity on a particular channel. As a result, decreased routing faults and channel switching can considerably reduce control packet transmissions, resulting in lower routing overhead.

Fig. 5(d) presents the performance comparison of SDRP with D^2CARP and CAODV in terms of throughput. Selecting the higher CAP is critical for increasing average throughput. In contrast, the D^2CARP and CAODV can not achieve a higher performance than SDRP in terms of throughput due to the random selection of idle channels. Channel's stability and quality play an essential role in sending the data packets correctly.

5.3.2. Performance of SDRP for reliable communication

In this section, the performance comparison of SDRP presented and analyzed with D^2CARP and CAODV as the number of CUs increases, the number of PUs is set to 10 with the PUs' activities time given by 200s and CBR traffic is generated over TCP with data rate = 0.54 Mbps. TCP ensures reliable communication.

In this experiment, we considered CBR to model the data traffic over

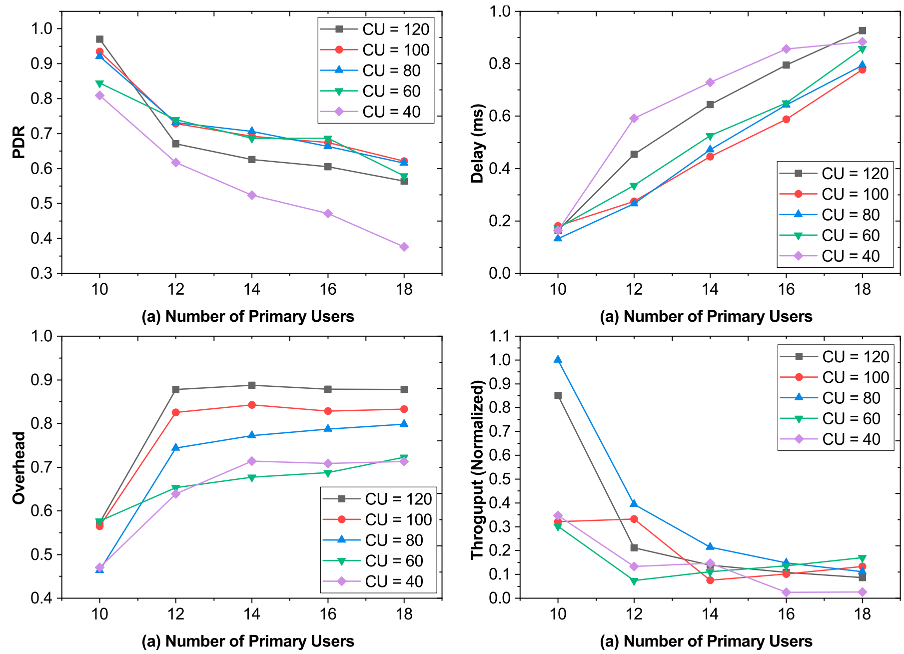

Fig. 11. The impact of multiple channels for PUs vs CUs on the performance behaviors for Scenario VII.

the Transmission Control Protocol (TCP). Consequently, all CU nodes will send data at the same bit rate. From Fig. 6(a), it reveals that when the number of CUs (CU = 40 or CU = 60) is low, PDR is high. In comparison, the PDR performance achieves a lower value up to approximately 75% of delivered packets for a higher number of CUs. TCP reliability requires a secure environment connection in CRNs associated with PU interference, link-error, and path/channel switching to avoid the TCP re-transmission connection. The growing number of CUs and mobility may raise the risk of interference and channel switching, resulting in additional TCP re-transmissions. Accordingly, SDRP will be suffering to satisfy TCP connection policies. However, SDRP can provide an acceptable PDR rate compared to D^2CARP and CAODV. Fig. 6(b) demonstrates that the SDRP achieves a significant improvement in delay compared to D^2CARP and CAODV. Each CU node is impacted by the PUs' activities for the lower number of CUs and is usually isolated due to a lack of idle channels. Therefore, the delay harvests at a higher rate with fewer nodes. Fig. 6(c) demonstrates the overhead rate for SDRP, D^2CARP and CAODV. TCP produces maximum traffic over reliable connections and decreases the amount of traffic over less reliable links. In contrast, CBR provides traffic irrespective of connections, thereby increasing the overhead traffic. Nevertheless, it is evident from the results that SDRP achieves lower overhead as compared to D^2CARP and CAODV. It is shown in Fig. 6(d) that SDRP achieves a significant improvement in terms of throughput as compared to D^2CARP and CAODV. It is observed that the relation is inversely proportional between throughput and delay.

5.3.3. Performance of SDRP with respect to PU diversity

In this section, performance evaluation of SDRP is compared and analyzed along with D^2CARP and CAODV with increasing number of PUs, number of CUs set to 100, PU activity time 200s and CBR traffic is generated over UDP with data rate = 0.54 Mbps.

From Fig. 7(a), it is observed that with the lower number of PUs = 10, the PDR is higher. However, as the number of PUs increases, the PDR reduces. It is due to the free number of licensed channels. Fig. 7(b) shows that the delay increases with the increasing number of PUs. This is because the rise in the number of PUs increases the uncertainty of channels. However, SDRP achieves lower delay than other compared protocols because the SDRP method combines higher CAP, which is responsible for substantially minimizing channel switching. Fig. 7(c) presents the performance comparison in terms of delay. Since the network is considered with PUs = 18, the overhead in all cases is high (approximately 85%). However, it is observed that the overhead in SDRP is lower than those in D^2CARP and CAODV. It can be seen by studying how SDRP deals with the PU's arrival over a particular channel. Because of the dynamic utilization of various stable paths and higher CAP, the probability of requesting a new route and channel during data transmission is lower, which reduces the SDRP overhead concerning CAODV and D^2CARP.

As illustrated in Fig. 7(d), the SDRP outperforms the D^2CARP and CAODV in terms of throughput rate as the number of PUs grows. With the increasing the number of PU, the throughput rate reduces because the PU is active in the current CU communication channel. Thus, the CU node needs to vacate the current CU communication channel with more route control overhead. However, it is noted that the SDRP outperforms its counterparts in terms of performance.

5.3.4. Performance of SDRP at large network scenario

In this set of experiments, the performance of SDRP is compared and analyzed with D^2CARP and CAODV. The number of PUs increased by an interval of 4, the number of CUs is set to 100 and CBR traffic is generate over UDP with data rate = 0.54 Mbps.

According to Fig. 8, PDR achieves a higher rate properly (95%), with the lower number of PUs set to 6, whereas PDR gradually decreases with an increasing number of PUs. The increasing number of PUs brings challenges in establishing a stable path and causes packets to drop. However, SDRP outperforms when compared to D^2CARP and CAODV. For the same reason, Fig. 8(a) shows that increasing the number of PUs results in changeling related to establishing a stable path out of PU activity. As a result, increasing the number of PUs creates a challenge in establishing a stable path out of effect PU activity, leading to increased packet drop. However, SDRP can outperform compared to D^2CARP and CAODV. SDRP acquires lower delays with a lower number of PUs (PU = 6, PU = 10), as shown in Fig. 8(b). However, SDRP acquires higher delay with a higher number of PUs between 14 and 22. The higher number of PUs causes CAP to fluctuate wildly between different quantities (e.g., 0.6, 0.7, 0.9, and upon). This increases the probability of channel switching compared to D^2CARP and CAODV, which selects an idle channel randomly without considering higher portability.

Fig. 4.9(c) demonstrates the overhead rate for SDRP, D^2CARP, and CAODV. Routing overhead increases with increasing network load due to varying PU numbers. Despite this, SDRP overhead gradually rises compared to D^2CARP and CAODV, where CARP and CAODV overhead continue to register higher rates. The SDRP constantly looks for the best channel, whereas higher PU numbers make finding the best channel difficult. Thus, the reason for the highest control packet overhead for SDRP is that it must always estimate the reliable channel, whereas D^2CARP and CAODV can choose any free channel randomly. Fig. 4.9(d) demonstrates the throughput rate for SDRP, D^2CARP and CAODV. The previous experiment's same aspects Fig. 4.8(d) remain, although the lower delay increases throughput. Similarly, when PU increases, it increases the delay and reduces the throughput. Fig. 8(b) demonstrates the overhead rate comparison for SDRP, D^2CARP and CAODV. Routing overhead is slightly higher with increasing the network load due to varying number of PUs. Despite this, the SDRP overhead progressively increases compared to D^2CARP and CAODV, where D^2CARP and CAODV overhead continues to register a higher rate. Fig. 8(d) demonstrates that the SDRP achieves higher throughput as compared to D^2CARP and CAODV.

5.3.5. Performance of SDRP with respect to channel diversity

The performance behavior of SDRP is compared and analyzed with D^2CARP and CAODV by increasing the number of channels, setting the number of PUs to 10, the PUs activity time 200s, the number of CUs to 50 and CBR traffic is generated over UDP with data rate = 0.54 Mbps.

From Fig. 9(a), as the number of channels decreased, the PDR rate for SDRP, D^2CARP, and CAODV decreased significantly. That is because of the chance of the idle channel decreasing due to PU activity. In contrast, an increasing number of channels will increase the opportunity to find free unused channels. However, SDRP can achieve higher PDR compared to D^2CARP and CAODV. The SDRP, D^2CARP, and CAODV suffer from a higher delay with fewer channels in Fig. 9(b). On the other hand, the delay rate reduces when the number of channels increases due to available free channels. On average, the delay rate for SDRP is lower compared to D^2CARP and CAODV. It is reasonable because SDRP always avoids the lower channel availability by updating the free channel list. Fig. 9(c) shows the relationship between overhead and the number of channels, where SDRP achieves lower overhead as compared to D^2CARP and CAODV. Fig. 9(d) demonstrates that the throughput rate achieved by SDRP is higher as compared to D^2CARP and CAODV. The lower number of channels limits the bandwidth, which reduces the throughput rate. On the other hand, the extra benefit is provided by increasing the number of channels that optimize the bandwidth, increasing the throughput rate. The advantage of a higher CAP enables SDRP to get higher rate throughput than D^2CARP and CAODV.

5.3.6. Performance of SDRP with respect to data rate diversity

In this set of experiments, The performance behavior of the selected routing protocols are analyzed with the increasing data rate, the number of PUs set to 10, the PUs activity time 200s and the number of CUs set to 100.

This scenario examines the effect of data rate on the performance of their protocols related to PDR in Fig. 10(a). MAC-layer inefficiencies in CRNs have been affected by increasing the data rate, reflecting on the routing performance. Therefore, the PDR rate reduces with increasing the data rate. For that reason, as predicted for reactive protocols, they are inappropriate for high data rates in CRNs. For the same reason, Fig. 10(b) shows that when the transmission rate increases, the delays for SDPR, D^2CARP, and CAODV increase.Fig. 10(c) explains the relationship between data rate and overhead. Increasing the data rate can boost to reduce overhead. This is because there is less data transfer (lower PDR), reducing the potential-routing error, which reduces overhead in CRNs. Fig. 10(d) reveals the relationship between the data rate and throughput for the three protocols. The throughput rate reduces when the data rate increases due to increasing spectrum bandwidth consumed by the data rate payload. However, the SDRP can adapt to increase the data rate to achieve an acceptable performance compared to D^2CARP and CAODV.

We examine the effect of various data rates on the presented protocols in Fig. 10(a). MAC-layer inefficiencies in CRNs is affected by increasing data rate, which in result affects the routing performance. Also, increasing the data rate reduces the range in which the signal is spread. Therefore, the PDR rate reduces with increasing the data rate. Fig. 10(b) presents the delay for SDRP, D^2CARP and CAODV increases when increasing the data rate increases. Fig. 10(c) demonstrates that the overhead decreases as the data rate increases. It is observed in Fig. 10(d) reveals that the throughput reduces when the data rate increases due to increasing spectrum bandwidth consumed by data rate payload. However, the SDRP can adapt to increasing data rate to achieve an acceptable performance compared to D^2CARP and CAODV.

5.3.7. Performance of SDRP at multiple PU on a channel (MPC) scenario

The performance behavior of SDRP is analyzed in this section with the number of PUs set to 10, the PUs activity time 200s, the number of CUs between 40 and 120. and CBR traffic is generated over UDP with data rate = 0.54 Mbps.

From Figure Fig. 11(a) related to the PDR perspective, the performance of SDRP under a diverse number of PUs and CUs is examined. In this experiment, we applied the MPC scenario, where the number of CUs and PUs gradually increased at a fixed channel number. The SDRP shows significant improvement when PU = 10 and CU = 120. It is reasonable due to a higher number of CUs that have a chance for intermediate CU nodes using paths outside of the PU region, which will increase the successful packet delivery, respectively. The lower PUs can exceed the idle spectrum/time, enabling SDRP routes to persist for extended periods as long as a reliable channel link is selected. In contrast, the average PDR moves down with the increasing PUs numbers and CUs diversity. It is reasonable due to the growing number of PUs with a static channel, making it tricky to find a free idle channel with a long-time PU activity. On the other hand, the channel-link assignment may fluctuate in the presence of a heterogeneous channel condition that drives inconsistency in the routing performance. Due to fluctuating.

Due to fluctuating channel conditions, the delay results in Figure Fig. 11(b) and overhead Figure Fig. 11(c) suffer for the same reasons. Thus, the cross-layer method enables the SDRP to select a channel with the highest CAP to avoid interference with the PUs and create a robust MCRNs routing protocol. Since there are so few CUs, each CU is affected by the PUs activity, as shown in Figure Fig. 11(d). As a result of the scarcity of available free channels, CU is frequently isolated. Thus, the destination CU cannot receive the data packet, reducing the throughput data rate. Failure to select a suitable channel route and find an alternate channel route can increase the delay and result in the lowest throughput. On the other hand, when CUs are set to 80, 120, the SDRP wins the highest bit data transfer rate. However, increasing PU density affects the throughput rate, which makes the SDRP achieve lower throughput values.

6. Conclusion and future work

In order to ensure a reliable route during the routing update period, a successful routing protocol requires a good channel and path selection strategy. This study has addressed the routing problem considering the difficult scenarios characterized by estimating and selecting the better idle channel availability under the assumptions of CUs mobility and channels availability especially when the PUs activity is time-variant. To address these challenges, a new cross-layer routing protocol called SDRP is proposed with a new channel selection approach for discovering and selecting a channel with a higher probability of channel availability. The SDRP updates its routing entries at every route update period for the next-hop assessment based on the approximate details of idle channels from the physical-layer and cross-layer interaction. The proposed protocol's efficiency is compared to the two benchmark protocols, i.e., D^2CARP and CAODV. Network simulation results show that SDRP has the desirable features of integrating channel availability probability (CAP) for selecting the best channel. It is also evident from simulation results that the SDRP achieves reduced routing overhead, end-to-end latency and improved average network throughput. Moreover, the SDRP outperforms other presented routing protocols in simulation scenarios with time-varying CUs and PUs, where more than one PUs could concurrently share the same channel, benefiting from the cross-layer paradigm and CAP optimal channel multi-hop transmission. We will explore the cross-layer routing protocol with higher layers as part of our future work to illustrate the Quality of Service (QoS) specifications for emerging applications. Furthermore, this work can also be extended to study routing using higher node mobility to enhance the available spectrum opportunity and routing stability. In addition, the smart applications foray lead to increased data rate transfer over multi-path/multi-channel, resulting in increased node energy consumption.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] M.A. Rahman, A.T. Asyhari, M.Z.A. Bhuiyan, Q.M. Salih, K.Z.B. Zamli, L-CAQ: joint link-oriented channel-availability and channel-quality based channel selection for mobile cognitive radio networks, J. Netw. Comput. Appl. 113 (2018) 26–35.

[2] S. Tyagi, S. Tanwar, N. Kumar, J.J. Rodrigues, Cognitive radio-based clustering for opportunistic shared spectrum access to enhance lifetime of wireless sensor network, Pervasive Mob. Comput. 22 (2015) 90–112.

[3] A. Guirguis, F. Digham, K.G. Seddik, M. Ibrahim, K.A. Harras, M. Youssef, Primary user-aware optimal discovery routing for cognitive radio networks, IEEE Trans. Mobile Comput. 18 (1) (2018) 193–206.

[4] M.A. Rahman, A.T. Asyhari, O.W. Wen, H. Ajra, Y. Ahmed, F. Anwar, Effective combining of feature selection techniques for machine learning-enabled iot intrusion detection, Multimed. Tool. Appl. (2021) 1–19.

[5] M. Rahim, M. Rahman, M.A. Rahman, A.J.M. Muzahid, S.F. Kamarulzaman, et al., A framework of iot-enabled vehicular noise intensity monitoring system for smart city, in: International Conference on Innovative Technology, Engineering and Science, Springer, 2020, pp. 194–205.

[6] R. Abozariba, M.K. Naeem, M. Patwary, M. Seyedebrahimi, P. Bull, A. Aneiba, Noma-based resource allocation and mobility enhancement framework for iot in next generation cellular networks, IEEE Access 7 (2019) 29158–29172.

[7] M.A. Rahman, M. Caleffi, L. Paura, Joint path and spectrum diversity in cognitive radio ad-hoc networks, EURASIP J. Wirel. Commun. Netw. 2012 (1) (2012) 1–9.

[8] Y. Saleem, F. Salim, M.H. Rehmani, Routing and channel selection from cognitive radio network's perspective: a survey, Comput. Electr. Eng. 42 (2015) 117–134.

[9] S. AlQahtani, A. Alotaibi, A route stability-based multipath QoS routing protocol in cognitive radio ad hoc networks, Wireless Network 25 (5) (2019) 2931–2951.

[10] R.N. Raj, A. Nayak, M.S. Kumar, QoS-aware routing protocol for cognitive radio ad hoc networks, Ad Hoc Netw. 113 (2021), 102386.

[11] M. Shakeri, N. Ardalani, P. Derakhshan-Barjoei, Improvement of network throughput by providing CAODV-based routing algorithm in cognitive radio ad hoc networks, Wireless Pers. Commun. 113 (2020) 893–903.

[12] C. Singhal, A. Rajesh, Review on cross-layer design for cognitive ad-hoc and sensor network, IET Commun. 14 (6) (2020) 897–909.

[13] A.S. Cacciapuoti, M. Caleffi, L. Paura, M.A. Rahman, Channel availability for mobile cognitive radio networks, J. Netw. Comput. Appl. 47 (2015) 131–136.

[14] T. Camp, J. Boleng, V. Davies, A survey of mobility models for ad hoc network research, Wireless Commun. Mobile Comput. 2 (5) (2002) 483–502.

[15] S. Salim, S. Moh, On-demand routing protocols for cognitive radio ad hoc networks, EURASIP J. Wirel. Commun. Netw. (1) (2013) 1–10, 2013.

[16] A. Kliks, D. Triantafyllopoulou, L. De Nardis, O. Holland, L. Gavrilovska, A. Bantouna, Cross-layer analysis in cognitive radio—context identification and decision making aspects, IEEE Trans. Cognit. Commun. Netw. 1 (4) (2015) 450–463.

[17] R. Saifan, A.E. Kamal, Y. Guan, A cross-layer routing protocol (CLRP) for cognitive radio network, in: 2013 IEEE Global Communications Conference (GLOBECOM), IEEE, 2013, pp. 896–901.

[18] A.S. Cacciapuoti, M. Caleffi, L. Paura, Reactive routing for mobile cognitive radio ad hoc networks, Ad Hoc Netw. 10 (5) (2012) 803–815.

[19] I.F. Akyildiz, W.-Y. Lee, K.R. Chowdhury, CRAHNs: cognitive radio ad hoc networks, Ad Hoc Netw. 7 (5) (2009) 810–836.

[20] S. Lavanya, M.A. Bhagyaveni, Design of sop based cross-layered opportunistic routing protocol for CR ad-hoc networks, Wireless Pers. Commun. 96 (4) (2017) 6543–6556.

[21] N. Dutta, H.K.D. Sarma, A probability based stable routing for cognitive radio adhoc networks, Wireless Network 23 (1) (2017) 65–78.

[22] A.S. Cacciapuoti, M. Caleffi, F. Marino, L. Paura, Routing update period in cognitive radio ad hoc networks, in: 2013 IEEE International Workshop on Measurements & Networking (M&N), IEEE, 2013, pp. 100–104.

[23] J.S.P. Singh, M.K. Rai, CROP: cognitive radio routing protocol for link quality channel diverse cognitive networks, J. Netw. Comput. Appl. 104 (2018) 48–60.

[24] S. Amiri-Doomari, G. Mirjalily, J. Abouei, Stability-based routing, link scheduling and channel assignment in cognitive radio mobile ad-hoc networks, Wireless Network 25 (4) (2019) 2013–2026, https://doi.org/10.1007/s11276-018-1793-1.

[25] R.N. Raj, A. Nayak, M.S. Kumar, Spectrum-aware cross-layered routing protocol for cognitive radio ad hoc networks, Comput. Commun. 164 (2020) 249–260.

[26] J. Ramkumar, R. Vadivel, Multi-adaptive routing protocol for internet of things based ad-hoc networks, Wireless Pers. Commun. (2021) 1–23.

[27] H. Dhingra, G.D. Dhand, R. Chawla, S. Gupta, An integrated service model to support user specific qos routing in cognitive radio ad hoc network, Peer-to-Peer Networking and Applications 14 (1) (2021) 18–29.

[28] A.R. Sangi, M.S. Alkatheiri, S. Anamalamudi, J. Liu, Cognitive aodv routing protocol with novel channel-route failure detection, Multimed. Tool. Appl. 79 (13) (2020) 8951–8968.

[29] A. Carena, M. Vaughn, R. Gaudino, M. Shell, D.J. Blumenthal, OPERA: an optical packet experimental routing architecture with label swapping capability, J. Lightwave Technol. 16 (12) (1998) 2135–2145.

[30] Y. Teng, M. Song, Cross-layer optimization and protocol analysis for cognitive ad hoc communications, IEEE Access 5 (2017) 18692–18706.

[31] Y. Du, L. Xue, Y. Xu, Z. Liu, An apprenticeship learning scheme based on expert demonstrations for cross-layer routing design in cognitive radio networks, AEU-Int. J. Electron. Commun. 107 (2019) 221–230.

[32] L. Gavrilovska, D. Denkovski, V. Rakovic, M. Angjelichinoski, Medium access control protocols in cognitive radio networks: overview and general classification, IEEE Communications Surveys & Tutorials 16 (4) (2014) 2092–2124.

[33] M. Cicioğlu, S. Cicioğlu, A. Çalhan, Sdn-enabled cognitive radio network architecture, IET Commun. 14 (18) (2020) 3153–3160.