FTCARP: A Fault-Tolerant Routing Protocol for Cognitive Radio Ad Hoc Networks

Zamree Che-aron[1], Aisha Hassan Abdalla[2], Khaizuran Abdullah[3] and Md. Arafatur Rahman[4]
[1,2,3] Department of Electrical and Computer Engineering,
International Islamic University Malaysia,
Kuala Lumpur 53100, Malaysia
[e-mail: [1]one_zamree@hotmail.com, [2]aisha@iium.edu.my, [3]khaizuran@iium.edu.my]
[4] Department of Biomedical Electronics and Telecommunications Engineering,
University of Naples Federico II,
Naples 80138, Italy
[e-mail: [4]arafatur.rahman@unina.it]
*Corresponding author: Zamree Che-aron

Received July 15, 2013; revised December 24, 2013; accepted January 15, 2014; published February 28, 2014

Abstract

Cognitive Radio (CR) has been recently proposed as a promising technology to remedy the problems of spectrum scarcity and spectrum underutilization by enabling unlicensed users to opportunistically utilize temporally unused licensed spectrums in a cautious manner. In Cognitive Radio Ad Hoc Networks (CRAHNs), data routing is one of the most challenging tasks since the channel availability and node mobility are unpredictable. Moreover, the network performance is severely degraded due to large numbers of path failures. In this paper, we propose the Fault-Tolerant Cognitive Ad-hoc Routing Protocol (FTCARP) to provide fast and efficient route recovery in presence of path failures during data delivery in CRAHNs. The protocol exploits the joint path and spectrum diversity to offer reliable communication and efficient spectrum usage over the networks. In the proposed protocol, a backup path is utilized in case a failure occurs over a primary transmission route. Different cause of a path failure will be handled by different route recovery mechanism. The protocol performance is compared with that of the Dual Diversity Cognitive Ad-hoc Routing Protocol (D2CARP). The simulation results obviously prove that FTCARP outperforms D2CARP in terms of throughput, packet loss, end-to-end delay and jitter in the high path-failure rate CRAHNs.

Keywords: cognitive radio ad hoc network, routing protocol, fault tolerance, path failure, route recovery

http://dx.doi.org/10.3837/tiis.2014.02.003

1. Introduction

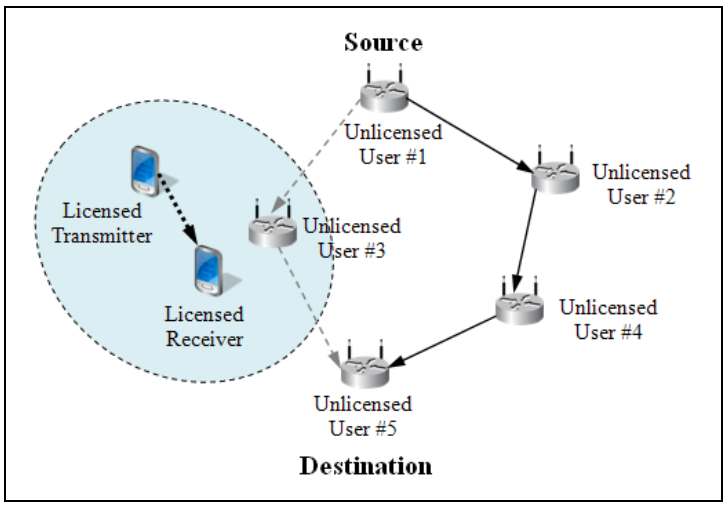
The rapid advancement of wireless technologies leads to a huge demand for radio spectrums, resulting in the spectrum scarcity problem. Furthermore, the traditional static spectrum management policy which assigns fixed bandwidths to licensed operators can cause the underutilization of radio spectrum [1]. Cognitive Radio (CR) [2][3] technology has been proposed as a standard for wireless communications in order to improve spectrum utilization and solve the problem of spectrum shortage by intelligently utilizing temporally unused licensed radio spectrums. By means of CR technology, unlicensed (secondary) users are allowed to opportunistically use the available spectrum portions which are underutilized by licensed (primary) users for data communication without harmful interference to them. Once a Primary User (PU) is active on a channel, all Secondary Users (SUs) in the PU's transmission range must immediately stop their activities on the channel.

In the paradigm of Cognitive Radio Ad Hoc Networks (CRAHNs), SUs can communicate with each other in ad hoc manner through both available licensed and unlicensed spectrum bands without relying on a preexisting infrastructure [4], resulting in the ubiquitous connectivity, minimal configuration, quick deployment and improved network scalability. The multi-hop communication is performed in case the source node cannot reach the destination node directly.

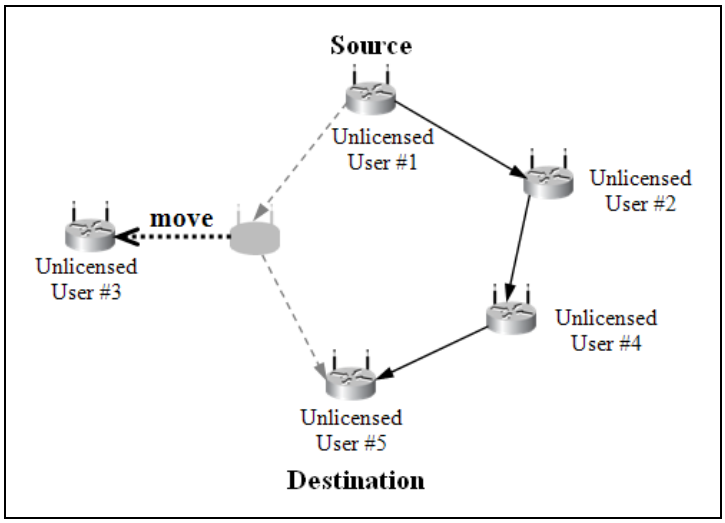
In fact, there are still many research open issues on CRAHNs that remain to be solved [5]. Routing constitutes one of the key technologies in CRAHNs and faces various significant challenges [6] that require in-depth studies. The major purpose of a routing protocol is to exchange up-to-date routing information and determine the appropriate path over which data is transmitted based on routing metrics as well as to discover a new path in case the current path is no longer available. One of the significant challenges on routing in CRAHNs involves the frequent changes of network topology, simply leading to route failures and service outages. The topological changes in CRAHNs occur primarily due to node mobility and intermittent PU activities, as shown in **Fig. 1(a)** and **Fig. 1(b)**. Since SUs have to instantaneously vacate the channel that overlaps a PU's transmission frequency once a PU activity is detected, the channel availability for each SU varies frequently. Furthermore, the dynamic spectrum usage can lead to the unstable and intermittent connectivity among SUs. Therefore, the issues of fault tolerance must be seriously considered in such networks. As mentioned above, unlike the classical wireless ad hoc networks, path failures in CRAHNs can result from not only faulty node, node mobility or link degradation but also PU activity. Different cause of a path failure may require different route recovery mechanism to handle it. With the unique characteristics of CRAHNs, the traditional ad hoc routing protocols (e.g. AODV [7], DSR [8], DSDV [9] and OLSR [10]) cannot be efficiently applied in such networks. Additionally, to the best of the authors' knowledge, recent research of cognitive ad hoc routing protocols pays little attention to the issues of fault tolerance in data communication.

In this paper, we propose a fault-tolerant routing protocol for CRAHNs, called the Fault-Tolerant Cognitive Ad-hoc Routing Protocol (FTCARP), which applies the joint path and spectrum diversity. The proposed protocol, which is an extension of AODV protocol [7], offers routing solutions and fault tolerance in CRAHNs. Our main goal is to provide reliable communication over the network and efficient route recovery mechanism in case of path failures occurring during data transmission caused by node mobility, PU activity, node failure, link degradation, etc. In FTCARP, a backup path is provided to all SUs which are forwarding

data packets towards the destination. During data delivery, if the SU encounters a route breakage, then its backup path will be immediately utilized in order to transfer the next coming data packets without transmission interruption. The simulation results show that the FTCARP protocol can increase the network throughput, reduce the number of dropped packets, decrease the data latency and achieve low data jitter in presence of path failures.



(a) PU activity



(b) Node mobility

Fig. 1. Challenges on data routing in CRAHNs

The rest of the paper is organized as follows. In Section 2, we review the related works. The proposed protocol is explained in Section 3. Section 4 describes the simulation model and parameters. Then, simulation results and evaluation are presented in Section 5. Finally, we conclude the paper in Section 6.

2. Related Work

Although some recent literatures study on routing protocols in CRAHNs, various issues on routing are still largely unexplored. In [11], a tree-based protocol has been proposed for CRAHNs based on assumption that all nodes are fixed or move very slowly. Furthermore, if a SU's data transmission is interrupted by a PU activity on a particular channel, all SUs must vacate that channel, even though they are not in the PU's transmission range. In [12], Chowdhury and Felice proposed a spectrum aware routing protocol for cognitive scenarios based on geographic routing paradigm, i.e. each SU can determine the location of other nodes, to elude regions of PU activity during the route formation. However, the route recovery approach, in case of path failures occurring during data delivery, is not taken into account in this protocol. Moreover, the impact of node mobility has not been evaluated. The authors in [13] has presented a cluster-based routing protocol, called the united node (UNITED), in order to minimize the end-to-end delay and maximize the network throughput in mobile CRAHNs. The SUs operates autonomously in a distributed manner and are grouped into a number of clusters. Nevertheless, the protocol may create a non-optimal (longer) path in situation that the source and destination node are placed near each other but in different clusters since data traffic must be forwarded to the cluster head of both clusters before arriving at the destination. In addition, the cluster heads consume much more energy than cluster members, resulting in the untimely shutdown to those nodes. The article in [14] has introduced the Cognitive Ad-hoc On-demand Distance Vector (CAODV) routing protocol, which applies individually path and spectrum diversity, with an aim to support dynamic CRAHNs. However, the network performance can be significantly degraded due to the impact of PU activity and node mobility because the protocol has not jointly considered path and spectrum diversity. In [15], the Dual Diversity Cognitive Ad-hoc Routing Protocol (D2CARP) has been proposed by sharing some common functionalities with CAODV. The protocol exploits the joint path and spectrum diversity to reduce the impact of performance degradation experienced by SUs because of PU activities. Nevertheless, since an efficient route recovery mechanism to cope with path failures caused by various problems in CRAHNs, e.g. node mobility, link degradation, node failure, etc., is not presented in the protocol, a large amount of packet loss may occur. Consequently, the network may suffer from a long service outage before a new path is discovered.

3. Fault-Tolerant Cognitive Ad-Hoc Routing Protocol

The Fault-Tolerant Cognitive Ad-hoc Routing Protocol (FTCARP) is a reactive distance-vector routing protocol proposed to deal with path breakages occurring during data transmission in CRAHNs, which shares some common features with D2CARP protocol [15]. The sequence numbers that can indicate the freshness of routing information are utilized to avoid the routing loop problem. The protocol jointly exploits path and spectrum diversity in data routing. By jointly utilizing both diversities, SUs can switch among different paths and channels for data communication in presence of frequency and space varying PU activity. It also takes advantage of the multiple available channels to improve the network performance. In addition, the novel fault-tolerant algorithm is provided to respond to route breakages in a timely manner by creating a backup path for all SUs which are transmitting data packets towards the destination. When the SU gets failure to forward a data packet through the primary route, it instantaneously exploits its backup path for transmitting the next coming data packets without any interruption. The backup path is selected based on the cause of path failure, i.e. PU activity, node mobility or link degradation. For different cause of a path failure, the protocol

will call different route recovery mechanism to handle it. By exploiting the fault-tolerant scheme, the data communication still keeps running continually in presence of route breakages. In FTCARP, each SU maintains two routing tables: Primary routing table and Backup routing table. Each routing entry records the IP address of destination node, the IP address of next-hop node for data forwarding, the channel number through which a data packet will be forwarded, and other relevant routing information. The basic operations of the FTCARP protocol include primary route discovery, backup path establishment and route maintenance which are described below in further detail.

In our CRAHN model, we assume that each SU is equipped with multiple wireless interfaces. Each interface can operate only on one of non-overlapping channels. The locations of PUs are assumed to be unknown to SUs.

3.1 Primary Route Discovery

When a source node requires a path towards a destination node for data communication, it broadcasts a Primary Route REQuest (P-RREQ) packet to its neighbors through all its available channels (i.e. not occupied by a PU). For an intermediate node, if the first P-RREQ packet is received, then it creates a primary reverse route pointing to the previous node and records the channel through which the packet has been forwarded as well as rebroadcasting the packet through all its vacant channels. In case it receives an additional P-RREQ packet with the same sequence number from the same node but on different channel, it just records a primary reverse route through that channel without rebroadcasting the packet. In such a way, the intermediate node can establish multi-channel primary reverse routes. Afterwards, the intermediate node will update the record of primary reverse route only if it receives the P-RREQ packet with a higher sequence number or the same sequence number but lower hop count. The out-of-date P-RREQ packet received by a node will be discarded to prevent the routing loop problem.

If a P-RREQ packet reaches the destination node or an intermediate node which has a valid primary route towards the destination (i.e. an active primary route entry with the same or higher sequence number than the sequence number stored in the P-RREQ packet), it generates a Primary Route REPly (P-RREP) and sends it back to the previous node through the same channel that the P-RREQ packet has been transmitted. The destination node will not discard the further P-RREQ packets received from the same node but on different channels. An intermediate node which receives the first P-RREP packet creates a primary forward route pointing to the packet sender through the same channel that the packet has been received and then forward the copies of the packet over all its valid primary reverse routes with different vacant channels (i.e. not used by a PU) towards the source of the P-RREQ packet. Afterwards, if an intermediate node receives an extra P-RREP packet from the same sender but on different channel, it establishes a primary forward route for the channel and only forwards the packet on its primary reverse route through that channel. In this fashion, multi-channel primary forward routes will be created. In case an intermediate node receives a fresher or better P-RREP packet (which has a greater sequence number or the same sequence number with smaller hop count), the entry of its primary forward route will be updated.

Once the first P-RREP packet arrives at the source node, a primary forward route for the packet-received channel will be created and then it starts transmitting the data packets destined for the destination node over the primary forward route via that channel. The source node also does not ignore the additional P-RREP packets received from the same node but on different channels. The source and intermediate nodes store active primary forward routes with different available channels (i.e. not occupied by a PU) and randomly select one of them for

data communication. The flow chart describing the process of primary route discovery is exhibited in **Fig. 2**.

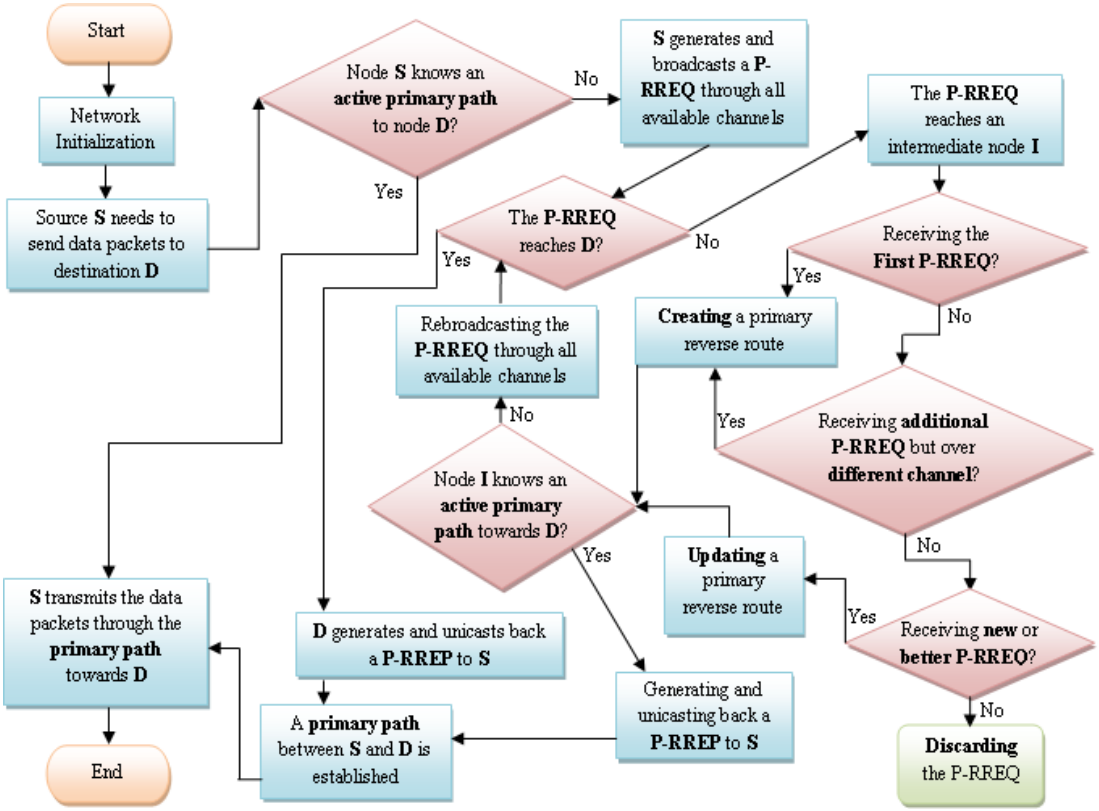


Fig. 2. Flow chart of primary route discovery process

As shown in **Fig. 3**, after the primary route discovery process is performed, the data traffic is delivered through the multi-channel primary path from the source to the destination ($1 \xrightarrow{ch2} 2 \xrightarrow{ch1} 8 \xrightarrow{ch2} 9 \xrightarrow{ch1} 10$).

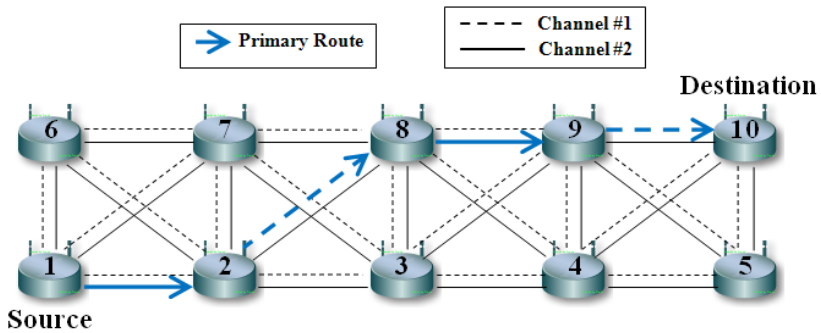


Fig. 3. Primary path establishment

3.2 Backup Path Establishment

After the primary path from the source node to the destination node has been established, the data delivery process will be triggered. The process of backup path establishment is performed during data transmission. Only SUs, except the destination node, that involve the data delivery process are able to create the backup path. Each of them is allowed to have only one multi-channel backup path towards a destination node. By ensuring these conditions, when a node on the primary path forwards a data packet to the next hop towards the destination, if its backup path towards the destination node has not been created, it broadcasts a Backup Route REQuest (B-RREQ) packet with limited TTL (Time-To-Live) value, for restricting the control packet flooding, to its neighbors through all its vacant channels. In the B-RREQ packet, some extra fields are added including "Data Source IP" field which contains the IP address of the data source, "Next Hop IP" field which stores the IP address of the next-hop node over the primary path, and "Primary Path Length" field which records the hop count of the primary path from the B-RREQ-originating node to the destination node. To avoid creating an ineffective backup path, the B-RREQ packet will be discarded in the following cases: 1) it is received by an intermediate node that has the same IP address as stored in the "Next Hop IP" field inside the packet; 2) it is arrived at an intermediate node whose next hop towards the destination is the B-RREQ-originating node; or 3) it reaches the B-RREQ source. In addition, the destination node ignores the B-RREQ packet if the hop count stored in the packet is equal to one. In case the B-RREQ packet is not discarded, a node responds to the first received B-RREQ packet by creating a backup reverse route pointing to the previous node through the channel that the packet has been received. The stored backup reverse route entry will be updated only if the node receives the B-RREQ packet with a greater sequence number or the same sequence number but lower hop count. The node is allowed to record only one lastly updated backup reverse route.

After processing the B-RREQ packet, an intermediate node may generate a Backup Route REPly (B-RREP) packet and send it back to the previous node through its backup reverse route if one of the following conditions is reached: 1) it has a fresh enough primary path towards the destination and the IP address of its next hop over the primary path towards the data source is the same as stored in the "Next Hop IP" field inside the packet; 2) it has a valid primary path towards the destination and the sum of the hop count of the valid primary path and the hop-count value stored in the packet is lower than the value of the "Primary Path Length" field inside the packet; or 3) it knows a valid backup path towards the destination and the sum of the hop count of the valid backup path and the hop-count value recorded in the packet is lesser than or equal to the value of the "Primary Path Length" field in the packet. Otherwise, it rebroadcasts the packet to its neighbors via all its available channels. In case the B-RREQ packet reaches the destination node, a B-RREP packet will be generated and forwarded back to the previous node via its backup reverse route only if its IP address is the same as recorded in the "Next Hop IP" field inside the packet or the IP address of its next hop over the primary path towards the data source is the same as stored in the "Next Hop IP" field of the packet.

The B-RREP packet will be forwarded back along the backup reverse path towards the B-RREQ-originating node. As the B-RREP packet travels back to the B-RREQ source, each intermediate node sets up a backup forward route pointing to the B-RREP sender via the same channel that the packet has come. The backup forward route will be updated if a further B-RREP packet which has a higher sequence number or the same sequence number with a lower hop count is received. After a B-RREP packet arrives at the B-RREQ source, its backup

path towards the destination node will be established. The flow chart explaining the process of backup path establishment is exhibited in **Fig. 4**.

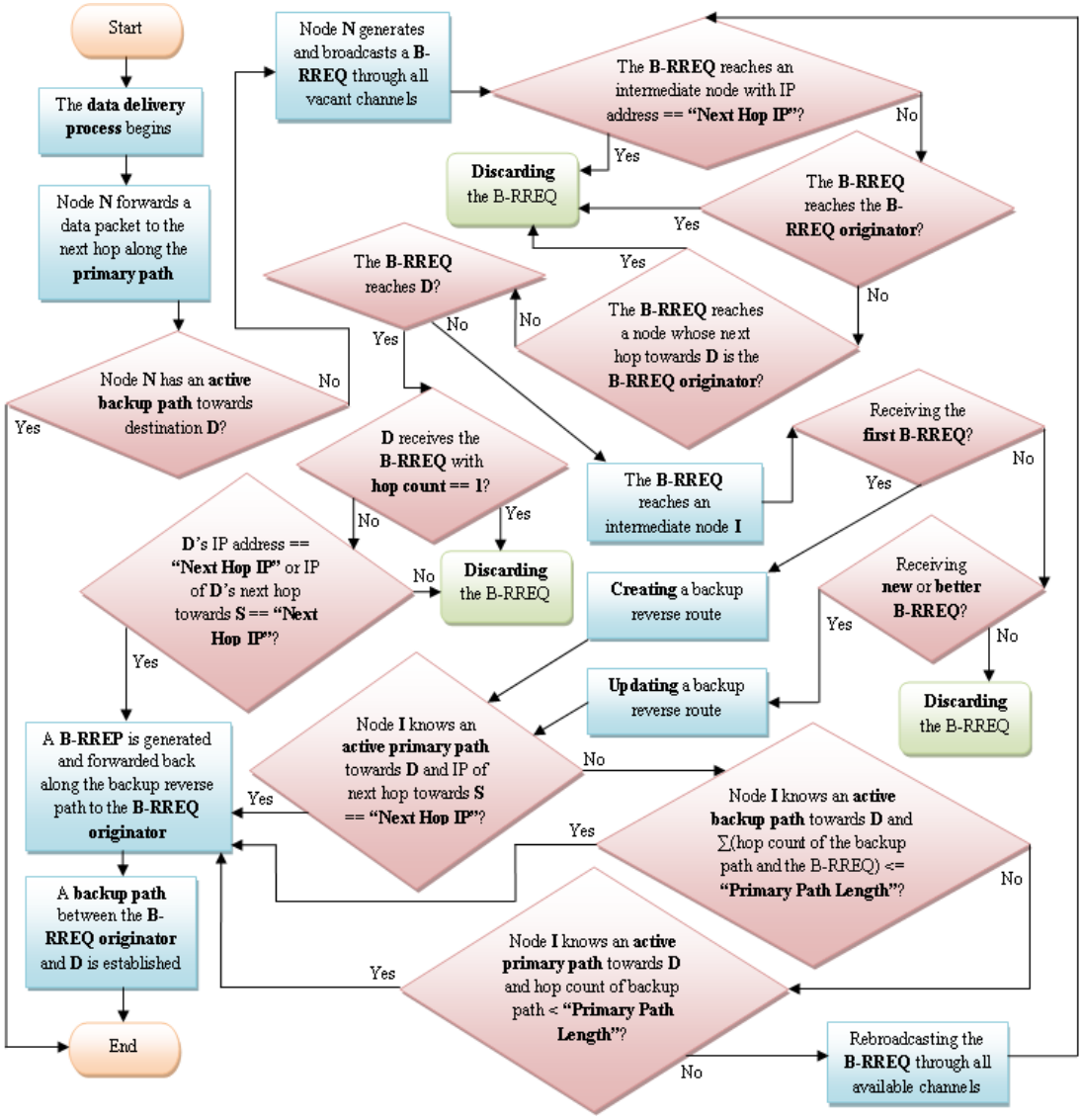


Fig. 4. Flow chart of backup path establishment process

As depicted in **Fig. 5**, the FTCARP protocol provides a multi-channel backup path to all nodes over the primary path (except the destination node). However, the data packets are still transmitted through the primary path ($1 \xrightarrow{ch2} 2 \xrightarrow{ch1} 8 \xrightarrow{ch2} 9 \xrightarrow{ch1} 10$).

3.3 Route Maintenance and Recovery

After a data packet has been successfully forwarded to a next-hop node over the primary path towards the destination, the lifetime of the primary forward route and the primary reverse route is increased in order to maintain the connectivity of primary path. Moreover, during data delivery, a timer is set for updating the backup links to preserve the connectivity information.

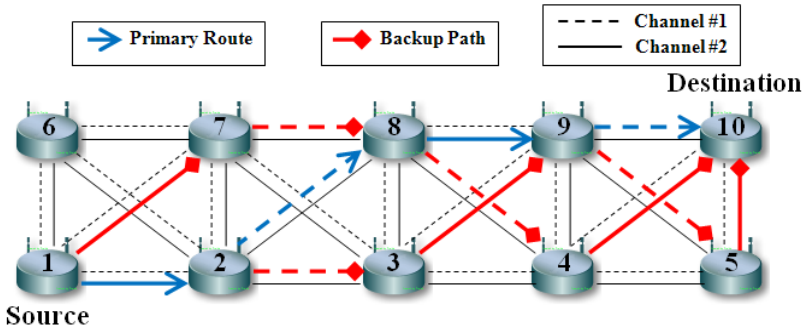


Fig. 5. Backup path establishment

In addition, a route breakage, which is detected using link-layer feedback, can be caused by node mobility, node fault, PU activity, etc. To provide fast and efficient route recovery, different cause of a path failure will be resolved by different route recovery mechanism. The flow chart shown in **Fig. 6** illustrates the route recovery mechanism of FTCARP protocol. During data transmission, when a node over the primary path detects a primary route breakage, it first checks the cause of the route failure. If the link breakage results from a PU activity, the node cannot transmit a data packet to the next-hop node via the currently used channel. Subsequently, it suddenly selects another available channel (i.e. not used by a PU) from its primary routing table to deliver next coming data packets to the same next-hop node towards the destination without changing the path direction.

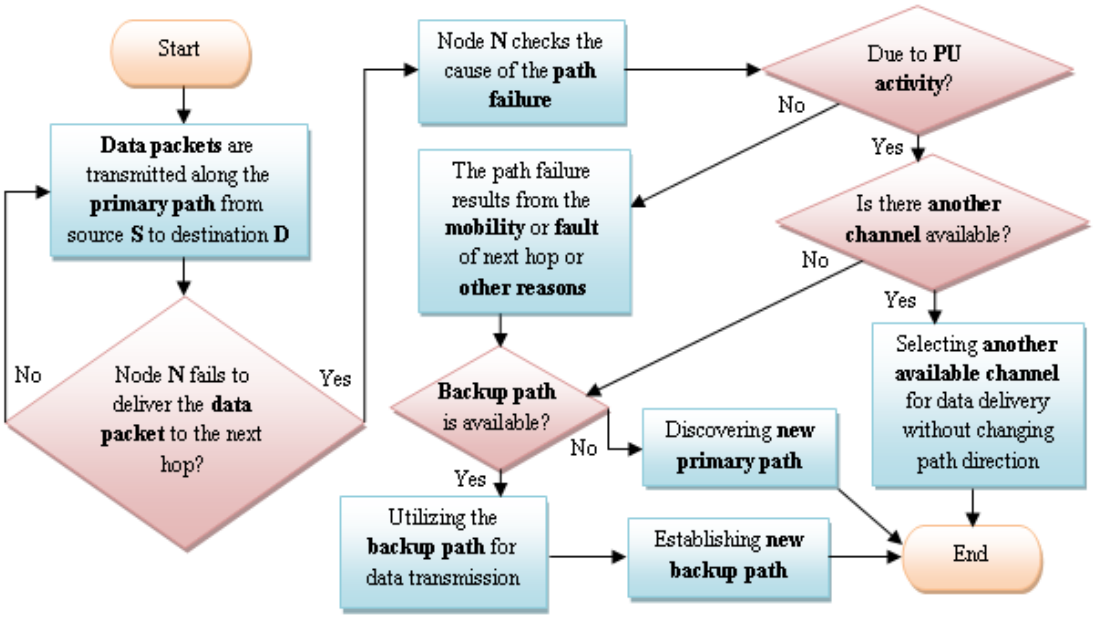
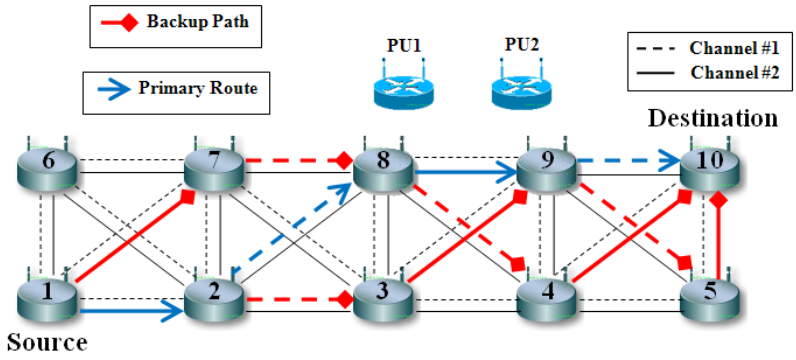


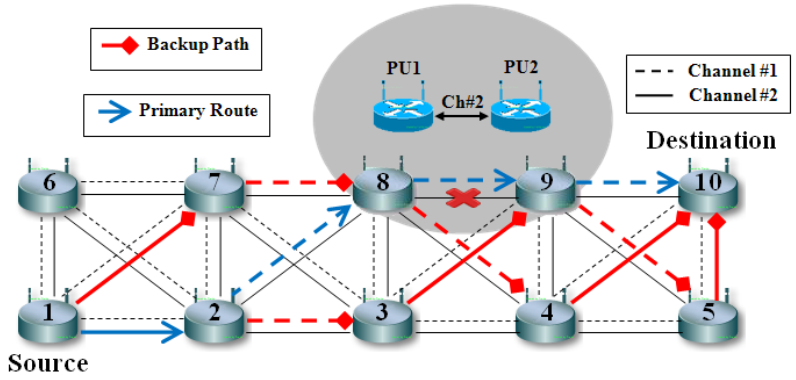
Fig. 6. Flow chart of route recovery mechanism

As exhibited in **Fig. 7(a)** and **Fig. 7(b)**, before a PU activity is detected, the data traffic is transmitted along the primary path ($1 \xrightarrow{ch2} 2 \xrightarrow{ch1} 8 \xrightarrow{ch2} 9 \xrightarrow{ch1} 10$). After PU1 starts communicating with PU2 through channel 2, node 8, which detects the PU activity, must immediately vacate its currently used channel which overlaps the PU's transmission frequency

(i.e. channel 2), resulting in the primary-path breakage. Node 8 also informs its neighbors (i.e. node 7, 2, 3, 4 and 9) about the channel unavailability via an error message. Subsequently, the neighbors disable their primary forward route over channel 2. As a result, node 2, 8 and 9 selects another available channel (i.e. channel 1), whose routing entry has been stored in their primary routing table, for transmitting the next coming data traffic by keeping the same path direction. Therefore, the new primary path ($1 \xrightarrow{ch2} 2 \xrightarrow{ch1} 8 \xrightarrow{ch1} 9 \xrightarrow{ch1} 10$) is still composed of the same nodes and has the same hop count.



(a) Before PU activity is detected



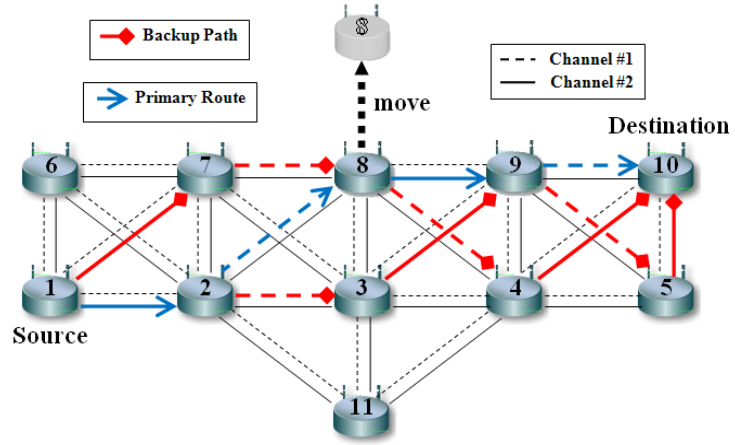
(b) After PU activity is detected

Fig. 7. Route recovery mechanism for a primary path failure caused by PU activity

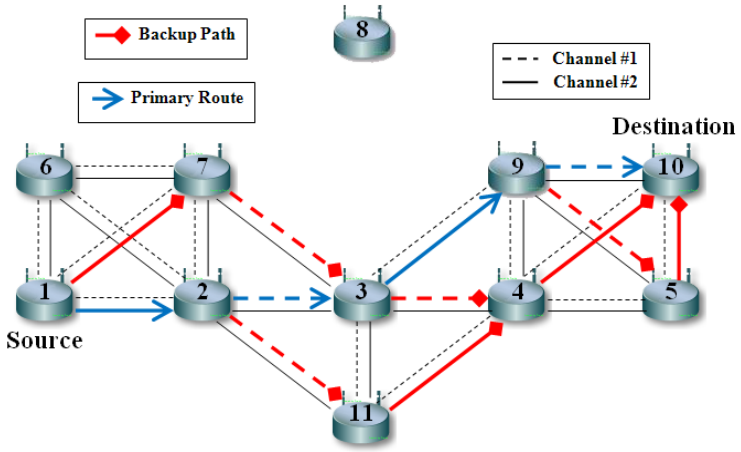
In the situation that a node cannot successfully forward a data packet to its next hop over the primary path due to the mobility or failure of the next-hop node, it instantaneously utilizes its backup path (i.e. its detour), whose routing entry has been kept in its backup routing table, to deliver the next coming data traffic without severe interruption of data transmission. After the backup path is exploited for data communication, the node establishes a new backup path instead of the currently used one. Subsequently, the node generates a Backup Route ERRor (B-RERR) packet containing the information of the backup path length (determined by hop count) and sends the packet to the source of data. An intermediate node over the primary path responds to the received the B-RERR packet by just forwarding the packet to the next hop towards the data source. When the B-RERR packet reaches the source node, it will discover a

new primary path for data transmission only if the hop count of the currently used backup path stored in the "Backup Path Length" field inside the packet is greater than the hop count of the broken primary path. The reason behind this is to prevent producing extremely high end-to-end packet delay.

As illustrated in **Fig. 8(a)** and **Fig. 8(b)**, before node 8 is moved, the data packets are delivered through the primary path (1–^{ch2}→2–^{ch1}→8–^{ch2}→9–^{ch1}→10). Once the position of node 8 is changed to which it is unreachable by node 2, the primary path is broken. When node 2 detects the path failure due to the mobility of node 8, node 2 instantaneously switches the transmission route to the backup path (2–^{ch1}→3–^{ch2}→9) for data delivery. Afterwards, node 2 informs the source node (i.e. node 1) of the hop count of the new primary path (1–^{ch2}→2–^{ch1}→3–^{ch2}→9–^{ch1}→10) through a B-RERR packet. Finally, the new backup routes will be created for node 1 (1–^{ch2}→7–^{ch1}→3), node 2 (2–^{ch1}→11–^{ch2}→4–^{ch2}→10) and node 3 (3–^{ch1}→4–^{ch2}→10).



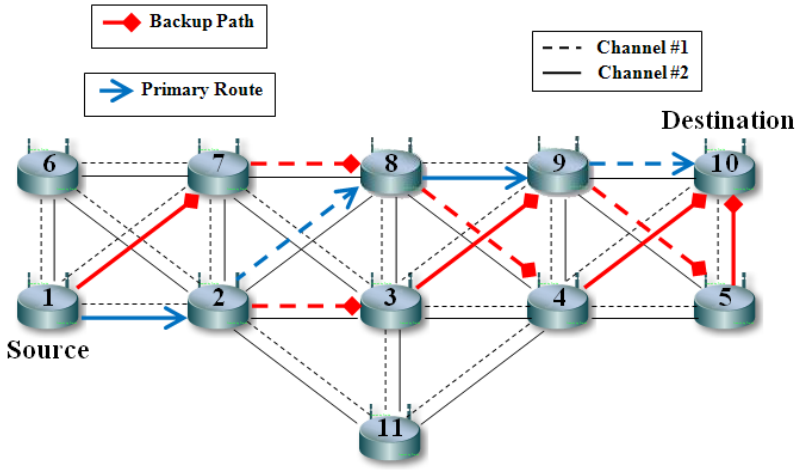
(a) Before node 8 is moved



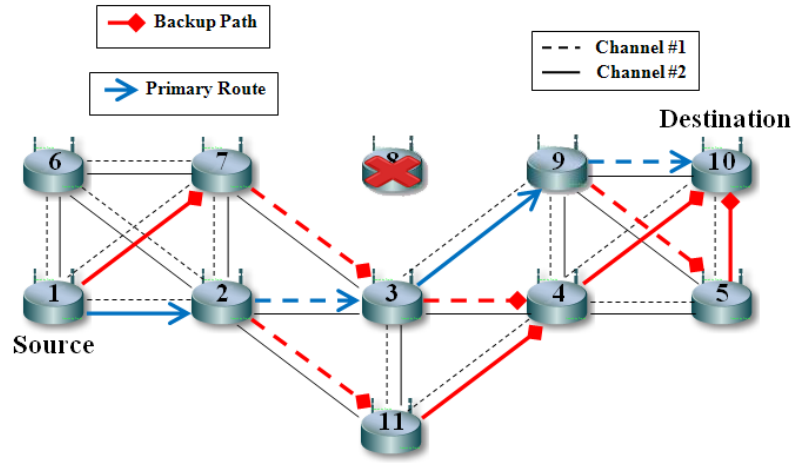
(b) After node 8 is moved

Fig. 8. Route recovery mechanism for a primary path failure caused by node mobility

Correspondingly, during data transmission, when node 2 detects the failure of node 8 (caused by any of a variety of reasons, e.g. broken node hardware, software bugs, or inadequate hardware resources), the affected data packets are suddenly forwarded through its backup path (2–^{ch1}→3–^{ch2}→9) to shorten the service disruption, as shown in **Fig. 9(a)** and **Fig. 9(b)**. Subsequently, the new backup paths for node 1 (1–^{ch2}→7–^{ch1}→3), node 2 (2–^{ch1}→11–^{ch2}→4–^{ch2}→10) and node 3 (3–^{ch1}→4–^{ch2}→10) will be discovered.



(a) Before node 8 is down



(b) After node 8 is down

Fig. 9. Route recovery mechanism for a primary path failure caused by node fault

4. Simulation Configuration

The efficiency of the FTCARP protocol is evaluated using NS-2 simulator [16]. We specify a simulation area of 1000 x 1000 m^2 in which 100 movable SUs and 10 fixed PUs are deployed. We model the PU activities according to the ON/OFF switching cycle. The ON state

represents the period where the channel is occupied by a PU and the OFF state denotes the period where the channel is available for SUs' communication. The UDP connections are created for the selected source-destination node pair. Over each UDP connection, CBR traffic with 512 byte data packets at the packet interval of 50 ms is transmitted. The simulation time is set to 400 seconds and the data transmission process is triggered after 10 seconds. There are 3 non-overlapping channels given for multi-channel data communications. The transmission range of the SUs is set to 150 m and the radius of interruption region for PU is fixed at 300 m. We use the IEEE 802.11 standard for MAC protocol. The two-ray ground reflection model is specified as the radio propagation type. The simulation parameters are summarized in **Table 1**.

Table 1. Simulation parameters

Parameter Names	Value
Simulation Field	1000 x 1000 m^2
Simulation Time	400 seconds
Data Start Time	10 seconds
Data Stop Time	400 seconds
Number of SUs	100
Number of PUs	10
Number of Channels	3
Traffic Type	CBR
Data Packet Size	512 bytes
Data Packet Interval	50 ms
MAC Layer	IEEE 802.11
Transport Layer	UDP
SU Transmission Range	150 m
PU Transmission Range	300 m
Radio Propagation	Two-Ray Ground Reflection

For our simulations, a path failure which occurs during data transmission results from either the movement of a SU (on the primary path) or a PU's activity. When a SU detects a PU's activity via a channel, the SU immediately invalidates all routing entries through such channel in its routing tables, even though the data delivery process is being operated.

5. Simulation Results and Evaluation

To evaluate the performance improvement, we consider the D2CARP protocol [15] as a benchmark for comparing with our proposed FTCARP protocol. The protocol performance is evaluated in the simulations with the different number of path failures occurring during data transmission. We use the NS2 Visual Trace Analyzer [17] to analyze the simulation results that are stored in the NS2 trace files. The performance metrics used for evaluation include: 1) average throughput: the average rate of data packets successfully received by the destination node per second; 2) packet loss: the total number of packets dropped during the simulation; 3) average end-to-end delay: the average latency time for all successfully transmitted data packets across a network from the source to the destination; 4) average jitter: the average of the variation in time between data packets arriving at the destination node.

In **Fig. 10**, we first compare the average throughput of FTCARP with that of D2CARP protocol under the different number of path failures. From the simulation results, the D2CARP protocol is clearly not able to cope with the networks encountering high path-failure rate by showing the large performance degradation in term of the average throughput, which drops

considerably when the number of path failures increases. On the other hand, the FTCARP protocol achieves higher average throughput compared to D2CARP. Since the fault-tolerant approach is applied in the FTCARP protocol, a breakage of primary path does not significantly affect the data delivery process.

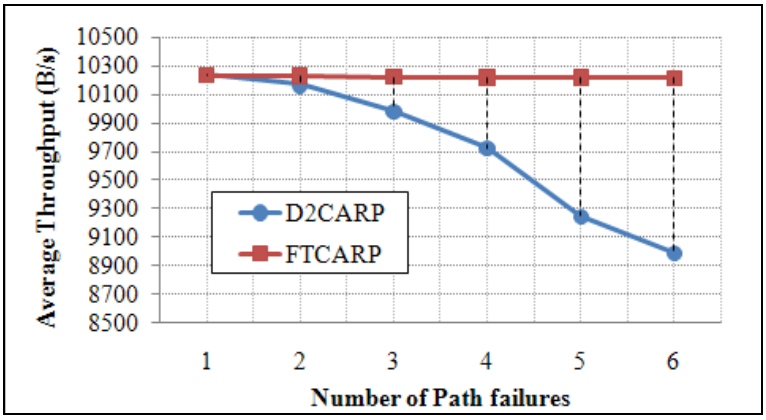


Fig. 10. Simulation results of average throughput

Fig. 11 shows the results of packet loss versus the different number of path failures. It is observed that the results of packet loss for the FTCARP protocol remain very low even under high path-failure rate conditions. As expected, without the efficient route recovery mechanism, the D2CARP protocol drops large numbers of data packets when a primary path breaks, especially due to node mobility. In such case, the data packets are still transmitted through the broken path until the source node is informed about the failure and a new primary path is discovered, leading to high packet loss in the network. In contrast, the FTCARP has stable performance even though the number of path failures rises.

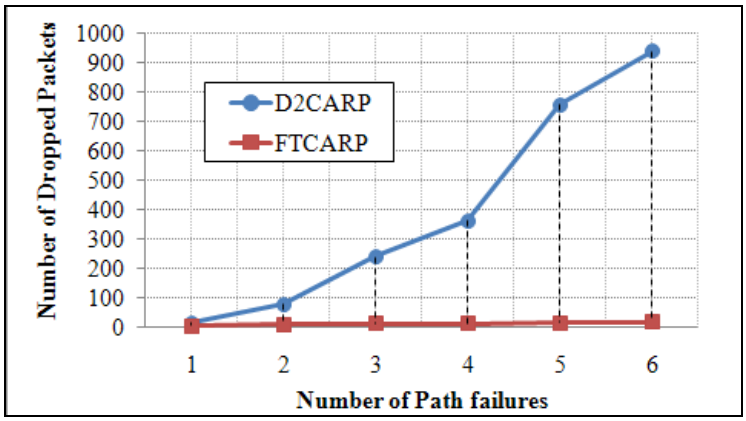


Fig. 11. Simulation results of packet loss

In **Fig. 12**, the simulation results of both protocols are evaluated in term of the average end-to-end delay against the increased number of path failures. We observe that the FTCARP protocol outperforms the D2CARP protocol in all cases due to its fast and efficient route recovery mechanism, whereas the D2CARP has worse performance. For FTCARP, when a primary path breaks during data delivery, a node over the path which encounters the failure

suddenly switches the transmission route to its backup path. On the other hand, D2CARP suffers from longer delay due to discovering a new transmission path.

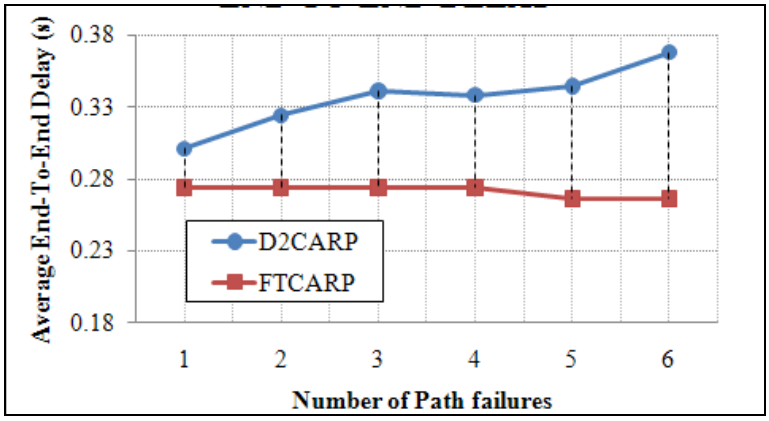


Fig. 12. Simulation results of average end-to-end delay

Fig. 13 exhibits the comparative results of average jitter as the number of path failures increases. It is obvious that, in the high path-failure rate networks, the FTCARP protocol provides lower average jitter as compared with that of the D2CARP protocol. The average jitter results for the D2CARP grow dramatically when the number of path failures increases. The fault-tolerant approach applied in FTCARP keeps the data transmission process still running continually even in presence of path breakages. On the contrary, for D2CARP, high path-failure rate greatly affects the network performance in term of jitter due to severe interruptions of data delivery, resulting in high average jitter results.

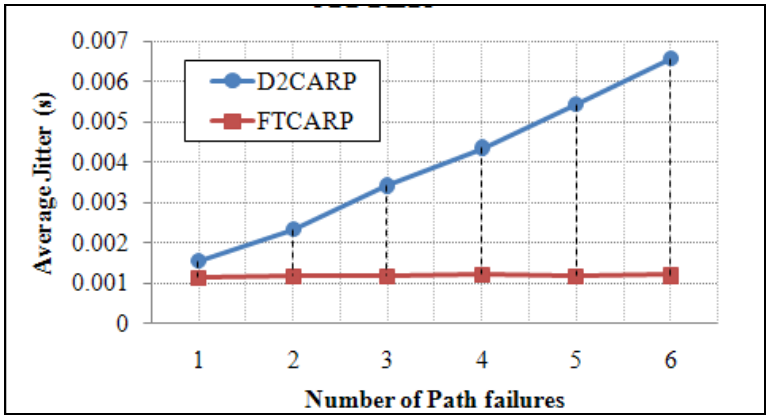


Fig. 13. Simulation results of average jitter

6. Conclusion

The CR technology holds great promises to cope with the spectrum insufficiency problem resulting from the increasing demands on radio spectrum due to the rapid advances in wireless communications technologies. In recent years, data routing in CRAHNs has received increasing research attention. However, it still faces numerous challenges and requires deep investigation. Previous research on CRAHN routing pays less attention to the issue of fault

tolerance.

In this article, we have proposed the FTCARP protocol, a fault-tolerant routing protocol for CRAHNs. The protocol jointly exploits the path and spectrum diversity for providing reliable communication and efficient use of spectrum in the networks. FTCARP can effectively cope with large numbers of path breakages occurring during data transmission by offering the fast and efficient route recovery mechanism. The protocol uses different route recovery mechanism to handle different cause of a path failure, i.e. PU activity, node mobility or faulty node. To evaluate the protocol effectiveness, the performance comparison between the proposed protocol and the D2CARP protocol has been conducted. As compared with D2CARP, the simulation results obviously prove that FTCARP provides higher average throughput, reduces average end-to-end delay, decreases average jitter, and achieves lower packet loss in the high path-failure rate CRAHNs.

Acknowledgment

This research project is based upon work supported by the Research Management Center (RMC) at our university. The authors would like to thank all members of the research group for their support.

References

[1] M. McHenry, "Spectrum white space measurements," in *Presentation to New America Foundation Broadband Forum*, June 20, 2003. Article (CrossRef Link).

[2] S. Haykin, "Cognitive radio: brain-empowered wireless communications," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 2, pp. 201-220, February, 2005. Article (CrossRef Link).

[3] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran and S. Mohanty, "Next generation /dynamic spectrum access/cognitive radio wireless networks: a survey," *Computer Networks*, vol. 50, no. 1, pp. 2127–2159, May, 2006. Article (CrossRef Link).

[4] I. F. Akyildiz, W.-Y. Lee and K. R. Chowdhury, "CRAHNs: cognitive radio ad hoc networks," *Ad Hoc Networks*, vol. 7, no. 5, pp. 810-836, July, 2009. Article (CrossRef Link).

[5] S. Sengupta and K. P. Subbalakshmi, "Open research issues in multi-hop cognitive radio networks," *IEEE Communications Magazine*, vol. 51, no. 4, pp. 168 – 176, April, 2013. Article (CrossRef Link).

[6] M. Cesana, F. Cuomo and E. Ekici, "Routing in cognitive radio networks: challenges and solutions," *Ad Hoc Networks*, vol. 9, no. 3, pp. 228-248, May, 2011. Article (CrossRef Link).

[7] C. E. Perkins, E. M. Belding-Royer and S. R. Das, "Ad hoc on-demand distance vector (AODV) routing," RFC 3561, Internet Engineering Task Force (IETF), July, 2003. Article (CrossRef Link).

[8] D. B. Johnson, Y.-C. Hu and D. A. Maltz, "The dynamic source routing protocol (DSR) for mobile ad hoc networks for IPv4," RCF 4728, Internet Engineering Task Force (IETF), February, 2007. Article (CrossRef Link).

[9] C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," *ACM SIGCOMM Computer Communication Review*, vol. 24, no. 4, pp. 234-244, October, 1994. Article (CrossRef Link).

[10] T. H. Clausen and P. Jacquet, "Optimized link state routing protocol (OLSR)," RFC 3626, Internet Engineering Task Force (IETF), October, 2003. Article (CrossRef Link).

[11] G.-M. Zhu, I. F. Akyildiz and G.-S. Kuo, "STOD-RP: a spectrum-tree based on-demand routing protocol for multi-hop cognitive radio networks," in *Proc. of IEEE Global Telecommunications Conf. (GLOBECOM)*, pp. 1-5, Nov. 30 – Dec. 4, 2008. Article (CrossRef Link).

[12] K. R. Chowdhury and M. D. Felice, "SEARCH: a routing protocol for mobile cognitive radio

ad-hoc networks," *Computer Communications*, vol. 32, no. 18, pp. 1983-1997, December, 2009. Article (CrossRef Link).
[13] A. C. Talay and D. T. Altilar, "UNITED nodes: cluster-based routing protocol for mobile cognitive radio networks," *IET Communications*, vol. 5, no. 15, pp. 2097-2105, October, 2011. Article (CrossRef Link).
[14] A. S. Cacciapuoti, C. Calcagno, M. Caleffi and L. Paura, "CAODV: routing in mobile ad-hoc cognitive radio networks," in *Proc. of 3rd IFIP Wireless Days Conf. (WD)*, pp. 1-5, October 20-22, 2010. Article (CrossRef Link).
[15] M. A. Rahman, M. Caleffi and L. Paura, "Joint path and spectrum diversity in cognitive radio ad-hoc networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2012, no. 1, pp. 1-9, July, 2012. Article (CrossRef Link).
[16] The VINT Project, "The network simulator - ns-2," 1995. Article (CrossRef Link).
[17] F. Rocha, "NS2 visual trace analyzer," 2012. Article (CrossRef Link).

Zamree Che-aron received his B.Eng. (First class honors) in Computer Engineering from Prince of Songkla University (PSU), Thailand, in 2007. He also obtained his M.Sc. in Computer and Information Engineering from International Islamic University Malaysia (IIUM) in 2010. Currently, he is pursuing a PhD in Computer and Information Engineering at International Islamic University Malaysia (IIUM). His main research interest lies in the areas of Wireless Communications, Broadband Network Technology, and Communication Protocols.

Aisha Hassan Abdalla received her B.Sc. and M.Sc. in Computer Engineering from University of Gezira, Sudan and her PhD in Computer Engineering from International Islamic University Malaysia (IIUM) in 2007. She was appointed as a lecturer in 1997. Currently she is working as Professor in Electrical and Computer Engineering Department, Faculty of Engineering at IIUM. Her current research interests include Data Communication and Computer Networking, ASIC Design, Computer Architecture and Grid Computing, Open Sources and Operating Systems. She published more than 100 papers in international journals and conferences. She is IEEE Senior Member, IEEE Women in Engineering Member and she works as a reviewer for many ISI journals.

Khaizuran Abdullah is an Assistant Professor in the Department of Electrical and Computer Engineering at International Islamic University Malaysia (IIUM). He received his Ph.D. in Electrical Engineering from RMIT University, Melbourne, Australia, in 2010. He also did his master degree in Electrical Engineering from Universiti Teknologi Malaysia (UTM), Skudai in 2003. Previously, he obtained his Bachelor of Science in Electrical Engineering from Ohio University, Athens, USA in 1997. Dr. Abdullah's main research interest lies in cognitive radio systems, signal processing communication, wireless communication particularly in Orthogonal Frequency Division Multiplexing (OFDM) techniques. Currently he serves as the Vice Chair of IEEE Communication Society (ComSoc) for Malaysia Section. He is also a registered member of Board of Engineers Malaysia.

Arafatur Rahman is working as a postdoctoral research fellow in the Department of Biomedical Electronics and Telecommunications Engineering (DIBET), University of Naples Federico II, Italy. He has completed his PhD degree from the same department. He has completed his master’s in Computer and Information Engineering from International Islamic University Malaysia (IIUM), Malaysia and graduation in Computer Science and Engineering from International Islamic University Chittagong, Bangladesh. He has received best student award in his master’s degree. He has also received fully funded government scholarship for his PhD. His research interests are mainly in wireless communication and cognitive radio networks.