

CR-CEA: A collision- and energy-aware routing method for cognitive radio wireless sensor networks

Rojin Tizvar · Maghsoud Abbaspour ·
Mahdi Dehghani

Published online: 1 May 2014
© Springer Science+Business Media New York 2014

Abstract Currently, working in the overcrowded shared unlicensed spectrum band, leads to a reduction in the quality of communications in wireless networks. This makes a considerable increase in packet loss caused by collisions that necessitates packets retransmissions. In the case of wireless sensor networks (WSN), a large amount of energy of sensor nodes will be wasted by these retransmissions. Cognitive radio technology makes it possible for sensor nodes, to opportunistically use licensed bands with better propagation characteristics and less congestion. In this paper a routing method for cognitive radio wireless sensor networks (CR-CEA) is presented, that is based on a cross-layer design that jointly considers route and spectrum selection. The CR-CEA method has two main phases: next hop selection and channel selection. The routing is performed hop-by-hop with local information and decisions, which are more compatible with sensor networks. Primary user activity and prevention from interference with them, is considered in all spectrum decisions. It uniformly distributes frequency channels between adjacent nodes, which lead to a local reduction in collision probability. This clearly affects energy consumption in all sensor nodes. In CR-CEA, route selection is energy-aware and a learning-based technique is used to reduce the packet delay in terms of hop-count. The simulation results reveal that by

applying cognitive radio technology to WSNs and selecting a proper operating channel, we can consciously decrease collision probability. This saves energy of sensor nodes and improves the network lifetime.

Keywords Wireless sensor networks · Cognitive radio · Energy aware routing · Cross-layer · Collision

1 Introduction

Wireless sensor networks (WSN) are composed of a group of inexpensive and tiny sensor nodes with limited resources that are able to communicate with each other through wireless media. When an event occurs in the sensing range of a sensor node, it collects information from the environment. Through the exchange of data between sensor nodes, the gathered information should reach to a central node named sink [1]. To accomplish this data transmission, the network requires a method for routing between these nodes. This method should be designed with regards to the specifications and limitations of sensor nodes. Energy is specifically one of the most important and limited resources of a sensor node. Hence, the energy consumption should be minimized as much as possible, with the goal of increasing network lifetime.

The major portion of energy consumption in sensor networks is due to data transmission between nodes. Since at the present time, most wireless applications are operating in shared unlicensed spectrum bands (2.4 GHz ISM band), this spectrum band is becoming overcrowded and congested. This leads to a reduction in the quality of communications in such networks [2]. In the case of WSNs, this quality reduction, that makes a considerable increase in packet loss and drop caused by collisions,

R. Tizvar · M. Abbaspour (✉) · M. Dehghani
Computer Engineering Department, Faculty of Electrical and
Computer Engineering, Shahid Beheshti University G.C.,
Evin, Tehran, Iran
e-mail: maghsoud@sbu.ac.ir

R. Tizvar
e-mail: r.tizvar@mail.sbu.ac.ir

M. Dehghani
e-mail: m.dehghani@mail.sbu.ac.ir

necessitate the packet retransmissions. This subject brings about a lot of sensor nodes energy to be wasted in these data retransmissions.

FCC investigations show that large numbers of frequency bands, which have been submitted permanently on the basis of allocation policies, are used only in limited regions and periods of time. Meanwhile, the mean utilization of these bands changes between 15 and 85 % [3]. Therefore, it appears that sharing licensed spectrum bands (such as TV broadcast: 54–806 MHz or cellular network) between unlicensed devices is an appropriate solution. The development in *Cognitive Radio* technology makes it possible to apply dynamic spectrum access model (DSA) in WSNs in order to have access to less congested spectrums possibly with better propagation characteristics. This technology allows unlicensed sensor nodes or secondary users (SU) to select temporarily the currently unused licensed spectrums, for their operation. In this process, the licensed users or primary users (PU) have no information about the existence of these SUs. This should cause no interference to the PUs operation [2].

The primary concept of cognitive radio presented by Mitola [4], was changed to the definition proposed by the FCC in this way: “A Cognitive Radio is a radio that can change its transmitter parameters based on interaction with the environment in which it operates [5]”. In other words, cognitive radio is a pattern for wireless communications, in which a network or a node has an opportunistic access to licensed bands. It reconfigures and changes its data transmission and receiving parameters based on its cognition of spectrum bands’ situation; parameters like operating frequency, transmission power and modulation technique. Thus, for SUs the possibility of using some portions of temporarily unused spectrums, which are idiomatically called Spectrum Holes or White Spaces, is provided by cognitive radio [6]. This reconfiguration is toward increasing the quality of communication, transmission and reception and improvement in channel utilization. A cognitive radio system has a main cognition cycle consisting of three main steps [7]:

- *Spectrum sensing*: A cognitive radio monitors the available spectrum bands, captures their information and then detects the spectrum holes,
- *Spectrum analysis*: The characteristics of the detected spectrum holes are estimated and
- *Spectrum decision*: the appropriate spectrum band is chosen according to the spectrum characteristics and user requirements and the system makes its reconfigurations.

It should be considered that the concept of “Cognitive Radio” was raised when the preliminary works on software defined radio (SDR) were being made [8]. SDR based RF

front-end transmitters and receivers are required for re-configurability of cognitive radio sensor nodes [9].

Enhancement in spectrum utilization and quality of communications, are the main benefits of using cognitive radio technology in wireless networks [9]. By applying cognitive radio features to WSNs, the main challenges in these networks are first of all: how and by which techniques the network should capture information from existing frequency bands? It should be considered that, due to the time varying nature of spectrum bands caused by PU activity, this spectrum sensing process should be done periodically. Secondly, what kind of information should be collected and which parameters should be analyzed? The Parameters can be PU activity, loss rate, delay and interference. Finally, according to spectrum sensing and analysis steps, how the entire network and each node should be reconfigured appropriately for further operation in this spectrum situation? It should also be kept in mind that all of these decisions must lead to meet the quality of service (QoS) requirements and primary limitations of a typical sensor network. There are some cognitive approaches being adopted by researchers to improve the performance of WSNs. In [10] some of these approaches such as schemes using the neural networks model and game theory are explained and compared.

Increasing the quality of communication prevents waste of energy resulting from a great number of packets retransmissions. This can be achieved by choosing a frequency band with better characteristics and reducing the probability of collisions and loss rates. A cross-layer approach that jointly considers route and spectrum selection is necessary for routing methods of cognitive radio wireless sensor networks [7]. This needs cooperation between network-layer and lower layers namely data link and physical layer. If a route from one SU (hop) to another leads to interference with a PU, this affects packet loss and delay in this route. To minimize this quality degradation various spectrum interfaces should be chosen in different hops in the same route. Hence a route from one source to a destination can probably include many intermediate nodes, which operate in different spectrum bands.

In this paper a new routing method for cognitive radio wireless sensor networks (CR-WSN) is presented that is based on a cross-layer design that jointly considers route and spectrum selection. This CR-WSN is clustered and cluster-heads that play a role as routing agents are responsible for performing cognition cycle. They select the proper sending channel at the network layer and the result is passed to the mac layer. In this method PU activity is considered in all spectrum decisions. Moreover, it uniformly distributes frequency bands between adjacent nodes, which lead to a local reduction in collision probability. This causes a great decrease in the number of

collisions on the entire network, which is explained to clearly affect energy consumption. Also, route selection is energy-aware and a learning-based technique is used to reduce the packet delay in terms of hop-count. In this technique by layering the CR-WSN an estimation of each cluster-head's distance to the sink is obtained. This estimation and the energy parameter are used by each hop to select the next hop.

The rest of this paper is organized as follows. In Sect. 2 some related works in the field of routing in cognitive radio Ad-Hoc and WSNs are described. The proposed method, its assumptions and related algorithms are explained in Sect. 3. In Sect. 4 the simulation results are presented. Finally, we conclude this paper in Sect. 5.

2 Related works

Many routing algorithms for common WSNs have been proposed that specifically focus on energy consumption and delay [11, 12]. However, predefined routing is not compatible with the dynamic behavior of cognitive radio networks, in which communication links and nodes connectivity change at different times. This is resulting from opportunistic spectrum access and its variable conditions. On the other hand, other routing approaches for cognitive radio Ad-Hoc networks are not completely adaptable for sensor networks, as there is no energy constraint in such networks. In this section we briefly explain some of these algorithms.

Routing algorithms for *cognitive radio Ad-Hoc networks* have been categorized into two main classes by Cesana and cooperators, depending on the assumptions taken on the issue of spectrum-awareness in proposed methods [13]. Some algorithms assume that network has full spectrum knowledge and spectrum occupancy map is available to all the network nodes or a central control entity. This class includes solutions such as graph-based algorithms. Secondly are routing schemes which base on local spectrum knowledge. They include all those solutions where information on spectrum availability is locally constructed at each SU through distributed protocols. In these algorithms, besides the computation of routing paths, the routing modules should be able to acquire network state information, such as currently available frequencies for communication, and exchange them with other network nodes [13]. It seems the distributed procedure of algorithms in second class, are more likely to be applicable in wireless sensor networks.

In [14] Ad-Hoc CR network is modeled as a layered and weighted graph. The authors assume that PUs are rarely active so that channels allocation and routing can be done statically. This network has different layers corresponding

to each channel. Horizontal edges are weighted by metrics like link bandwidth, availability and load, and vertical edges by cost of spectrum handoff and improvement in signal to noise ratio. The problem of routing is finding the shortest and the least cost path in this graph. Another solution is modeling by a colored graph, which is presented by Zhou and et al. [15]. This algorithm is centralized and is also suitable for multi-hop networks that PUs are infrequently active. Finding the maximum clique of a conflict graph, is another approach proposed in [16]. All possible routes between each pair of nodes and for each route all sorts of channel allocation are calculated in this approach.

The main goal of the algorithm presented by Pyo and Hasegawa [17], is to find routes with minimum cost. In this system cost of a link is defined by a function of transmission power. It should be noted that in this algorithm, interference with PUs and their activity are not considered. On the other side, in many systems transmission power metric is not enough to address the challenges of multi-hop cognitive radio networks.

Xie et al. [18] have proposed a method that aims the use of relaying to enhance the spectrum utilization. There are two multi-hop routing strategies, named NNR and FNR. They made a tradeoff between single and multi-hop routes of SUs and the predetermined interference threshold of PUs.

In addition to common delay metrics in routing algorithms such as hop count or queuing delay, spectrum band switching delays should be also taken into account in cognitive radio networks. Switching delay between frequency bands ($D_{\text{switching}}$) and medium access delay within a given frequency band (D_{backoff}), are considered in routing algorithms presented in [19] and [20]. These two metrics are generalized in [21, 22] where $D_{\text{switching}}$ and D_{backoff} are integrated with queuing delay in a node, which serves many incoming flows. These methods based on the adoption of an on-demand protocol that is a variation of AODV [23].

Some algorithms focus on throughput maximization. The spectrum aware mesh routing (SAMER) algorithm [24] opportunistically routes traffic across paths with higher spectrum availability and quality via a new routing metric namely path spectrum availability (PSA). It balances between long-term route stability and short-term opportunistic performance with the goal of improvement in spectrum utilization and total end-to-end performance. Channel bandwidth and loss rate are the main parameters considered in SAMER.

CRP is a distributed routing algorithm for CR Ad-Hoc networks [25]. Chowdhury and Akyildiz have considered 3 key CR-specific performance metrics: (1) *Explicit protection for PU receivers*, (2) *Two CR routing classes* and (3) *Scalable joint route-spectrum selection*. The route-setup is

composed of two steps; first spectrum selection and second next hop selection. The procedure of the algorithm is mainly based on RREQ and RREP messages (like AODV) on a common control channel (CCC), which is assumed to be 2.4 GHz unlicensed bands.

Here we explain two routing algorithms that have been specifically proposed for CR-WSNs, ERFLA [26] and ECR [27]. ERFLA tries to effectively use limited energy of a sensor node to improve the network lifetime. In this network there are some coordinators which are responsible for detecting white spaces. These nodes render the spectrum sensing results to spectrum broker which intimates every node to communicate in a particular band of spectrum which doesn't cause any harmful interference to the PUs. The network is leveled, clustered and sectorized. By an event occurrence, the source node floods data packets in the direction of sink. This algorithm decreases the number of flooding packets by making sensor nodes sleep and wake-up in different sectors. ECR is an energy-aware routing protocol that addresses the challenges in CR-WSN such as dynamic spectrum access and energy constraint. Two spectrum bands are assumed to be available: 2.4 GHz unlicensed ISM band and 680 MHz UHF licensed TV band. It is said that for the same transmission power, the maximum range in this licensed band is almost double the maximum range in the 2.4 GHz [28]. There are two types of nodes in the network: reduced function devices (RFD) and full function devices (FFD). FFDs that operate as cluster-heads, are responsible for spectrum sensing. This algorithm is also based on routing setup by RREQ and RREP messages. Nodes energy, number of common channels between them, hop count and availability of the licensed channel are main route selection metrics in this algorithm.

There are some issues that must be considered in protocol design for CR-WSNs. For example, PU activity awareness and prevention from SU interference with them are important issues that are not taken into account by some of explained algorithms. On the other hand, physical-layer quality metrics like interference, transmission power and spectrum switching delays are not sufficient for CR-WSNs. Furthermore, most of routing algorithms for Ad-Hoc CR networks, select the whole route and all the middle nodes from source to destination. Such approaches are not suitable for CR-WSN. Rather, solutions with distributed, local and hop by hop route and spectrum decisions are more compatible with sensor networks, due to energy, memory and computational constraints.

3 Proposed routing method

In this section a distributed routing method for CR-WSNs is proposed. This routing is performed hop-by-hop with

local information and decisions. The method is based on a cross-layer design that provides the ability to select a proper working channel in the network layer. Furthermore, it is a layered network, in which route selection is energy-aware and to decrease the overall delay of packets, a learning-based technique is used. This solution targets the reduction of collision occurrence probability which reduce retransmission in the network. This leads to less energy consumption, which is one of the main goals in a typical WSN and increase the lifetime of the network. Delay is also one of the parameters that should be considered in protocol designs for WSN to achieve better performance. In addition, the proposed method is distributed that is more efficient in WSN. Distributed methods with local decisions usually dissipate less power comparing to central methods with centralized decisions.

In the following the proposed routing method is described in detail. At first the assumptions have been cleared and then the network topology and model are described. At last the routing method is presented in detail.

3.1 Assumptions

It is assumed that the CR-WSN is a clustered and layered, which is composed of three types of nodes; each with a unique identifier (ID):

- *Sensors*: These nodes have a predefined communication range called r_s and are deployed randomly in the network region. They use a transceiver that can be adjusted to work in a range of different frequencies. Sensor nodes do not participate in routing.
- *Cluster-heads*: Spectrum sensing and collecting information from them is the main capability of these nodes. Same as sensor nodes, their transceivers can be adjusted to work in a range of different frequencies. Due to the fact that Cluster-heads have more tasks to perform than sensor nodes, their initial energy is higher. Their deployment is preplanned and they have a specific communication range shown by r_{ch} . Their responsibility is to route data messages from sensor nodes to the sink. Cluster-heads have also data aggregation [29] ability.
- *Sink*: It has sufficient initial energy and is the final destination of all data messages. Sink has knowledge about all cluster-heads locations. The r_{sink} denotes its communication range which covers the entire map.

It is worth noting all nodes in this network assumed to be immobile and have static locations. However, by applying a signaling protocol that informs moving sensors about their current cluster and subsequently its cluster-head, the mobility can be added to this method. Adding this signaling protocol causes no flaw in the main routing algorithm process.

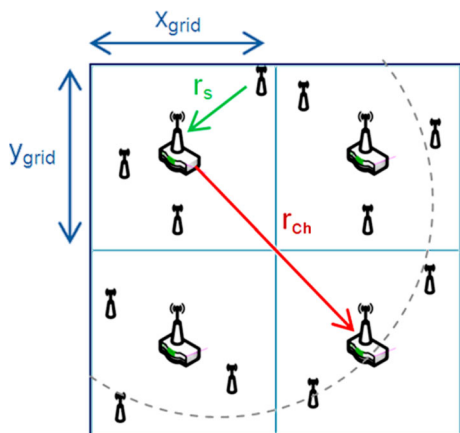


Fig. 1 Grids and communication ranges

3.2 CR-CEA network model

There is no limitation on the network map and clustering method. To easily describe the algorithms we assume the map is a rectangle and clustering is done by dividing the map into grids. It is worth noting that this clustering method can be replaced by other clustering methods in heterogeneous WSN.

3.2.1 Clustering

Each cluster-head is located in a grid center. The grid size and number of cluster-heads can be calculated as follows (Fig. 1):

$$x_{grid} = y_{grid} \leq \sqrt{2}r_s \tag{1}$$

$$r_{ch} = 2r_s + \varepsilon, \tag{2}$$

where r_{ch} and r_s are respectively the communication ranges of cluster-heads and sensor nodes. These equations guarantee the inter-cluster and intra-cluster connectivity. Number of cluster-heads is calculated as follows:

$$\left(\frac{X_{map}}{x_{grid}}\right) \times \left(\frac{Y_{map}}{y_{grid}}\right) = \frac{X_{map} \times Y_{map}}{2r_s^2} \tag{3}$$

after cluster-heads deployment, each sensor node should be notified about its cluster-head. In this phase, each cluster-head broadcasts a control message (PT_ChCtl: its format is described in the next section). After receiving first message of this type, a sensor node considers the sender as its cluster-head and saves its ID.

3.2.2 Layering

During the initialization phase of the network, sink computes number of layers according to the width of layers

($Width_L$). Then, corresponding to each layer creates and sends a control message (PT_SCtl). Sink adjusts its transmission power proportional to its distance from the layer, so that each cluster-head in that layer receive this message. Obviously, the first received message of this type is used by a cluster-head to determine its layer, and save its *id*. Sink locates in L0, the nearest layer to sink is L1 and they are identified consecutively by L2, ..., LN. Besides *id*, each layer has an attribute, named Layer-Cost that will be described subsequently in the algorithm. Number of layers is calculated as follows:

$$\begin{aligned} \text{Number of Layers} &= N \\ &= \left\lceil \frac{\text{Max}(\text{ClusterHead Distance from sink})}{Width_L} \right\rceil \end{aligned} \tag{4}$$

3.2.3 Neighbor discovery

All neighbors of the cluster-head that receive the *sent control message* in clustering phase consider the sender as their neighbor; extract and save its characteristics from this message.

3.2.4 Cognitive radio model

There exist k licensed frequency bands (channel) C_1, C_2, \dots, C_k in this CR-WSN, in which PUs operate. The 2.4 GHz unlicensed band is shown by R_0 and is always available for SUs which are sensors, cluster-heads and sink. While at least one of k licensed frequency bands is available for SUs, R_0 is not selected as operating frequency. In addition, all control messages in network initialization phase, are sent through R_0 . It can be said that R_0 plays a role similar to CCC, which is mentioned before.

- *Spectrum sensing*: This process is performed at the physical layer and is relevant to structure of cognitive radio nodes (here cluster-heads), which is not within the scope of this paper. The comparison of sensing techniques that can be used in CR-WSN is presented in [9].
- *Spectrum analysis*: The node analyses the captured data from a spectrum sensing. At the end of this step the node determines required information for “spectrum decision” phase. This information is PU activity, which affects availability of channels and can be used to detect free channels and also the information about quality of channels such as probability of loss rate.
- *Spectrum decision*: As it is mentioned earlier about spectrum opportunistic access, in a CR network, a SU can operate in a specific spectrum just when its operation has no effect on PU activity. So, not to interfere with any PU is the most important issue in spectrum selection in this method. This step results to

the appropriate operating frequency. The details of this spectrum selection will be explained later.

Due to the dynamic behavior of PUs and spectrum situation, this cognition cycle should be performed periodically. Cluster-heads have this responsibility. On the other hand, this algorithm has two strategies for spectrum selection with a focus on: (1) presence of PU and spectrum availability with the goal of reducing the number of collisions, (2) priority of available spectrum based on their quality in addition to PU activity. In the next section, the algorithm is explained by the first strategy. The algorithm’s behavior in the second strategy and its results will be described in Sect. 4.

3.3 CR-CEA routing method

This method basically consists of three algorithms: route-cost estimation, cognition cycle channel selection and routing channel selection. Route-cost estimation algorithm is used with the aim of decreasing the delay of a route in proposed method. In a CR network, according to different situations of spectrum bands, appropriate channels should be selected in each network region, so that all the nodes can adjust their transceivers properly to work in a specific frequency band. At the end, the main routing algorithm, which employs these algorithms and the information obtained from the messages, will be described step by step. During the routing process, each hop determines the 2-tuple (N_{id}, C_{nx}) where N_{id} is next-hop ID and C_{nx} is the selected routing channel. In the following at first the message types and formats are described. Then the related algorithms are mentioned and at last the routing method has been shown in detail.

3.3.1 Format of messages

There are both control and data messages. Control messages are broadcast by the sink and cluster-heads. The final destination of all data messages that contain the information sensed by sensors is the sink. Figures 2 (a, b) and 3 (a, b) illustrate the format of these messages. The scheduling and process of control messages will be also explained later.

3.3.2 Route-cost estimation algorithm

In a WSN, number of visited hops (hop-count) is a reasonable metric for estimating the delay each packet experiences. Therefore, here hop-count is used for the cost of each layer. As previously mentioned, in this method each

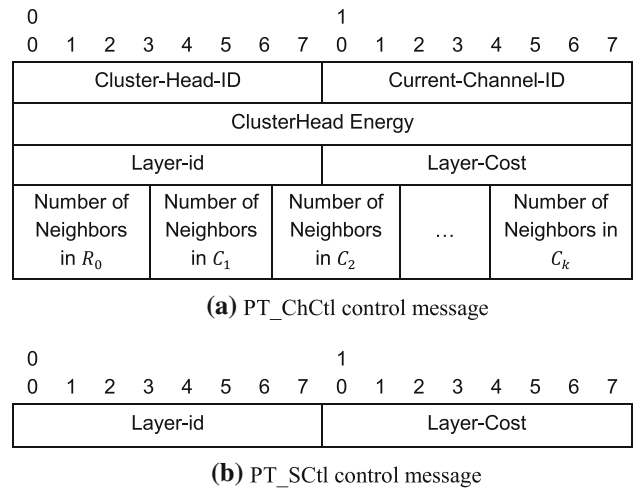


Fig. 2 CR-CEA control messages format

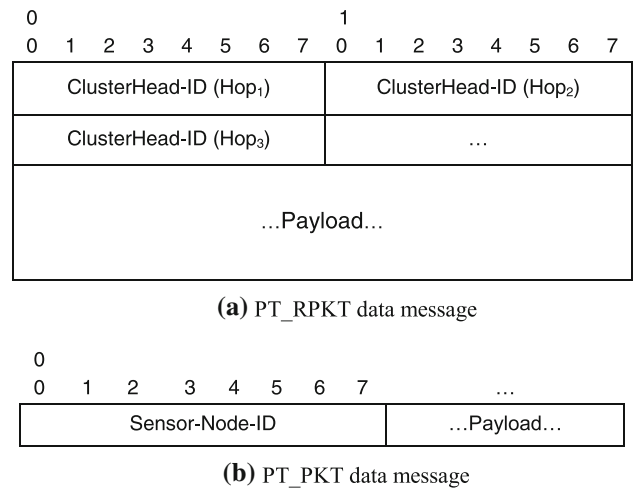


Fig. 3 CR-CEA data messages format

routing agent, here means cluster-head, locally decides on the next hop. If each cluster-head has an estimation of all its neighbors’ distances to the sink, making a route by next hop selections will lead to an overall better route. The distance is defined as the number of hops between a cluster-head and the sink. This estimation is included in the Layer—Cost parameter. Based on a learning process and using a weighted average, an algorithm called Cost_Calculation is used to calculate this parameter for each layer. Layer-Cost has an arbitrary initial value $(Cost_{x,0})$ that over time reaches to the actual cost of reception from each layer.

This algorithm runs in the sink and this node sends updated information to all cluster-heads. There are two variables named *CalculatedCost* and *CurrentCost* for each layer. They are initialized with the initial value of *Layer-*

```

Algorithm Cost_Calculation (Layer-id, HopCount):
1   $\tau \leftarrow \text{Defined\_Threshold}$ 
2   $\text{CalculatedCost} \leftarrow (\text{CalculatedCost} + \text{HopCount}) / 2$ 
3  if  $|\text{CalculatedCost} - \text{CurrentCost}| / \text{CurrentCost} > \tau$ 
4    then  $\text{CurrentCost} \leftarrow \text{CalculatedCost}$ 
5         $\text{CREATE\_PT\_Sctl}(\text{Layer}, \text{CurrentCost})$ 
6         $\text{SEND}(\text{PT\_Sctl})$ 
    
```

Fig. 4 Cost_Calculation algorithm in pseudocode

Cost. Sink extracts hop-count and ID of all visited hops from each PT_RPKT message it receives (Fig. 3a). Since this node has the map of the entire network, it can determine the layers that message has traversed. Then for each layer, it executes Cost_Calculation algorithm, which is displayed in Fig. 4 in pseudocode, and calculates the CurrentCost of the layer. Actually this algorithm plays a role similar to heuristic function $h(x)$ in path finding in a directed graph [30].

Due to the fact that next hop selection is performed by approximate knowledge of routes costs, there is no need to always have the exact value of them. Therefore, a threshold is defined to prevent the sink from sending PT_Sctl messages for all data messages it receives and this saves energy of the cluster-heads because they receive less control messages. The value of this threshold only affects the network warm-up time, which is the time that *Layer-Costs* get their final value and will no longer be updated. As it is illustrated on Fig. 4 the cost of layer x after receiving n th message from the nodes in this layer, which is shown by $Cost_{x,n}$, will be calculated by Eq. 5:

$$Cost_{x,n} = \frac{Cost_{x,n-1} + H_{x,n}}{2}, \tag{5}$$

where $H_{x,n}$ is the number of hops this message has visited and $Cost_{x,n-1}$ is the last calculated cost for this layer. The explicit form of Eq. 5 will be:

$$Cost_{x,n} = \frac{1}{2^n} \left(Cost_{x,0} + \sum_{i=1}^n \left(2^{(i-1)} \times H_{x,i} \right) \right), \tag{6}$$

according to Eq. 6, the limit of $Cost_{x,n}$ as n approaches infinity, is independent from $Cost_{x,0}$. So, as it is said before $Cost_{x,0}$ has an arbitrary initial value which only needs to be greater than the limit value.

3.3.3 Cognition cycle channel selection algorithm

An appropriate channel should be selected for sending the message from one node to another. If this channel is selected in such a way that packets encounter less

collisions, the reliability of the route will increase. Subsequently, it prevents many retransmissions and duplicate receptions which are done by transport layer protocols of the network.

As it is mentioned above, each cluster-head periodically senses C_1, C_2, \dots, C_k spectrums to be informed about frequency bands occupied with PUs (cognition cycle). If the results from the spectrum analysis step reveal that channel C_i is occupied by a PU, at that time C_i will be omitted from the choices. We assume n channels are not available because of PUs activity. In decision step, all these n channels will be omitted. As regards cluster-head knowledge about its one-hop neighbors, among $k-n$ remained channels, it selects the one, in which minimum number of neighbors are working. This selected channel that is now named as *Current-Channel*, is used by all sensor nodes in this cluster to communicate with the cluster-head. As they are notified by the cluster-head, they reconfigure their transceiver to work in this frequency band. Reduction of collision probability in sending packets from adjacent clusters is the reason of this procedure for channel selection.

In some cases the currently utilized channel is also available after recent spectrum sensing. If so, to reduce the overall switching delays ($D_{switching}$), it is preferred to stay in current channel. However, it is possible that after a period of time, most of the cluster-heads work in a limited number of channels, which will obviously increase the collision probability and the network will behave as a non-cognitive radio network. Therefore, a randomization technique is used for switching to another spectrum in lately described situations.

3.3.4 Routing channel selection algorithm

After next hop selection, an appropriate channel should be selected, through which the message is sent to this next hop (Nid, C_{nx}). In this method we mainly focus on reduction of collision occurrence probability. Simultaneous reception of packets from incoming flows of the same frequency band is the main cause of collisions. Accordingly, a channel should be selected, from which the destination node (next hop) has the minimum probability of packet reception. For example cluster head X wants to send a routing packet to next hop Y. According to this algorithm, to decrease the probability of collision occurrence in Y, X will send the packet to Y in an available channel that minimum number of Y's adjacent clusters work on. So X needs to know the number of Y's neighbors working in each channel. The required information was included in the last PT_Chctl received from this node. This information is the number of neighbors working in each C_i ($i = 1 \dots k$). Finally, among $k-n$ remained channels, the source cluster-head selects the channel, in

which the minimum number of next hop's neighbors work and sets C_{nx} as it.

It is worth noting that while at least one of C_1, C_2, \dots, C_k is available, R_0 is not returned as a result of these two algorithms. On the other hand, after channel selection and before any transmission the sender and the receiver perform a channel coordination at the mac layer.

3.3.5 Messages scheduling

The scheduling of this algorithm has two independent phases. One is related to exchanging control messages and information extraction which is done periodically, another is routing a data message sent by a sensor to the sink, which happens obviously after an event occurrence.

As long as a cluster-head has sufficient energy to work, at intervals of τ seconds (local time), it senses the spectrum bands and selects proper channel by *Cognition Cycle channel Selection algorithm*. Then, after receiving a message of any type it creates a PT_ChCtl message and fills all its fields in. Now it broadcasts this message on Current_Channel. This message is sent even if the value of Current_Channel field is not changed after spectrum sensing. The reason is that all neighbors need to know the updated value of cluster-head energy and number of its other active neighbors in each channel. They need this information for their next hop selection phase. The PT_ChCtl message is sent *after receiving a message*, in order to avoid from simultaneous sending of these messages by cluster-heads.

All nodes in the communication range of this cluster-head will receive this PT_ChCtl. The sensor that receives this message from its own cluster-head, extracts Current-Channel-ID and reconfigures its transceiver for working in this new channel. From now on, the sensor will send all PT_PKT messages through this channel. The cluster-heads that receive this PT_ChCtl, extract, update and save all the

information in this message, if the sender is one of their neighbors. Figure 5 illustrates the flowchart of this message scheduling.

After an event occurrence, a data message is created and routed (by the routing protocol described in the next section) to the sink. The sink executes the Cost_Calculation algorithm, after it receives a data message. Then for each layer that its cost is changed, this node creates a PT_SCTl message. Afterwards, it broadcasts this PT_SCTl on R_0 . Due to the fact that R_0 is always available in the entire network; this spectrum is selected for sink control messages.

When a cluster-head receives this PT_SCTl message, if its own layer matches the Layer-id in the message, it updates the cost of its layer to the new Layer-Cost value. Furthermore, if one of its neighbors locates in this layer, it also updates the corresponding field saved for this neighbor. This means the costs of neighbors' layers is updated in two ways; first by receiving a PT_ChCtl from the neighbor itself, second by PT_SCTl sent by the sink. As a result if any of these messages don't reach to the destination, the cluster-heads can obtain the last updated values from the other message; so they almost always have a good estimation of cost of the different routes toward the sink. The flowchart in Fig. 6 presnets this part.

3.3.6 CR-CEA routing algorithm

In this method instead of selecting the whole route and all its middle nodes from one source (cluster-head) to the destination (sink), each hop in the route locally selects the next hop. To complete the data route from a source to the destination, each hop determines the 2-tuple $\langle NId, C_{nx} \rangle$ where NId is next-hop ID and C_{nx} is the selected channel. In fact, this routing algorithm has two phases first next hop selection, which is done according to QoS parameters including energy and estimated delay. Second, for this

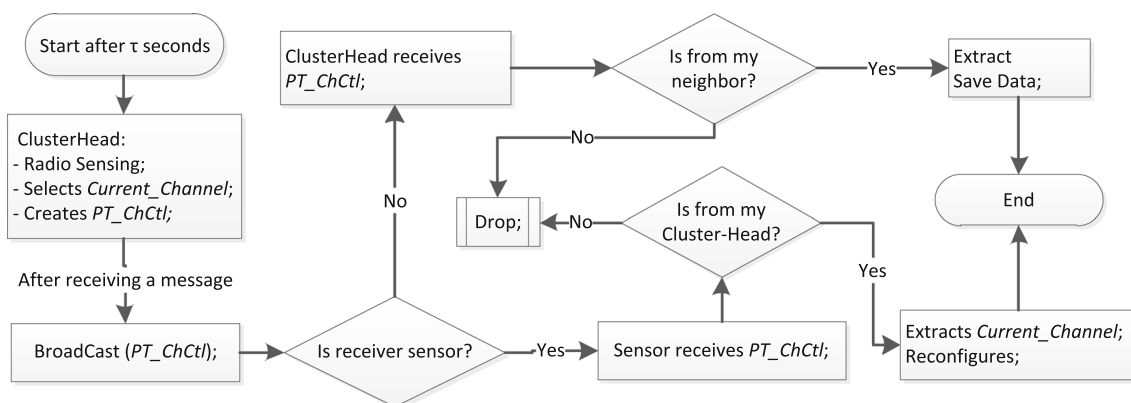


Fig. 5 Flowchart of periodical phase of message scheduling

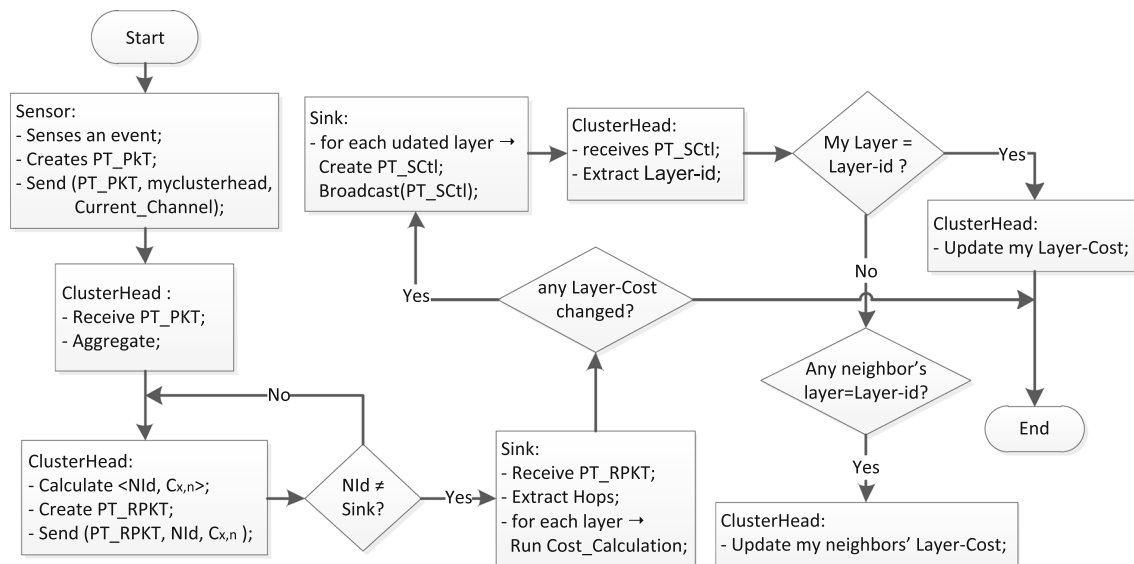


Fig. 6 Flowchart of CR-CEA routing steps and PT_Sctl control message scheduling

selected next hop, an appropriate channel should be selected, through which the message is sent to the next hop.

Thus far, all the routing method requirements including messages and algorithms have been described in details. Here the process of routing a data message, after an event occurrence till it reaches to the sink is explained step by step:

1. When an event occurs, each sensor around it creates a PT_PKT message and puts the sensing information in the message payload; then, sends this PT_PKT on Current-Channel to its cluster-head.
2. After cluster-head receives these messages, performs the aggregation process and then creates a PT_RPKT message which contains the final results of sensing information. It also appends its ID to the header of the message. Now, as it is described before, this node should select a $\langle Nid, C_{nx} \rangle$.
3. For each neighbor, it calculates the value of W using the Eq. 7:

$$W_{(Nid)} = k_{(Nid)} \cdot \frac{Layer - Cost_{(Nid)}}{Energy_{(Nid)}}, \quad (7)$$

where $k_{(Nid)}$ is a coefficient which for each neighbor has a different value. This parameter is used to prioritize the neighbors, to meet the specific network requirements. In this method the neighbor which is closer to the sink has more priority to be selected. This concept is included in the Layer-id. Hence, here $k_{(Nid)}$ equals to neighbor's Layer-id. The neighbor that has the minimum W will be selected as next hop (Nid). As we choose a greater value for Width_L parameter, means fewer layers, the number of neighbors in the

same layer with the same Layer-id and Layer-Cost increases. As a result, in this step of the routing, the effect of neighbors' energy will increase. Therefore there is a better load balancing between neighbors. On the contrary, with narrower layers, the neighbor which is closer to the sink is mostly selected as the next hop.

Obviously for those cluster-heads that have sink in their neighbors, $W_{(sink)}$ is considered as 0. Now Nid is selected; the cluster-head executes the Routing Channel Selection algorithm with Nid argument as its input. This algorithm results to the proper C_{nx} .

4. The cluster-head sends the PT_RPKT to next-hop (Nid) through C_{nx} .
5. When the next hop receives this PT_RPKT, appends its ID and select the next Nid, C_{nx} .
6. Steps 4 and 5 continue until the message reaches to the sink.
7. At last sink executes the Cost_Calculation algorithm and transmits the result to all cluster-heads.

The flowchart in Fig. 6 completely presents the steps of routing method.

4 Performance evaluation

We have implemented CR-CEA method in the NS-2.31 simulator [31]. For adding cognitive radio capability to the nodes in this simulator, we applied the changes proposed in [32] and exploit the *Cognitive Radio Cognitive Network Simulator* patch presented in [33]. By using this patch the routing algorithm in the routing agent is able to determine

through which network interface and channel the packets should be sent. How different network layers are implemented in this simulation is briefly shown in Table 1. The network's behavior is investigated, in different traffic loads. As it can be seen in the table, we have used 4 uniformly generated traffic models L1, L2, L3 and L4. Traffic model L1 models a situation in the network, in which on each second ten events happen in the network and will be sensed by sensors around it. This model simulates a low traffic load in the network. Similarly, L2 and L3 respectively simulate medium and heavy traffic loads, by generating 100 and 1,000 events per second. In addition, we applied bursty traffic load, by model L4, which generates 20 events per second in a single grid. The values assigned to methods' parameters in the simulation and the values of NS-2 parameters are shown respectively in Tables 2 and 3.

A set of exponential random functions are used to model PUs' activity and a parameter is defined to represent the channel availability ($Availability_{_}$). A scheme of the simulated CR-WSN topology is also illustrated in Fig. 7. This scheme is based on the network model presented in Sect. 3.2.

In this section the network behavior and CR-CEA routing method are evaluated in different situation and traffic loads. We have investigated criteria such as delivery and collision rates, energy consumption and overhead of the method. At the end this method is compared with ECR and AODV.

4.1 Delivery rate

In all parts of the simulation, the packets may be dropped because of these reasons: collision occurrence, TTL expiration, errors and the defined maximum hop-count which is also purposed to prevent packets to be lost in routing loops. Here, We have investigated the data message delivery rate in the network. Figure 8 illustrates the delivery rate of the network for different number of the licensed channels in

the network (k). The zero point in x -axis represents the situation in which only the unlicensed spectrum R_0 is available in the network; that simulates the case that the WSN in not cognitive radio. The diagram shows that increasing the number of channels is more effective when heavy traffic like $L3$ flows in the network.

Figure 9 shows instantaneous delivery rate of the network during the whole simulation time. The vertical dashed line in this diagram shows the time at which the network transits from the transient state to the steady state. In heavy traffic there is a reduction in delivery rate; that is caused by an increase in the number of dropped packets because of collisions.

As it can be observed in these two diagrams, by increasing the number of licensed channel from 0 to 8 the network behavior in medium traffic $L2$ gradually become

Table 2 Assigned values of CR-CEA parameters

Parameter	Value
X_{map}	500 m
Y_{map}	500 m
r_s	70 m
r_{ch}	150 m > $2r_s$
r_{sink}	550 m (NS default)
x_{grid}	100 m
y_{grid}	100 m
Number of Cluster-Heads	25
Number of Sensors	200
$Width_L$	$r_{ch}/2 = 75$ m
Number of Layers	7
k	{0, 1, ..., 8}
τ	1 s
$k_{(NId)}$	The Layer-id
$Cost_{x,0}$	$x = \{1, 2, \dots, 7\}$

Table 1 Implementation of all network layers

Layer	Model	Details
Application	Traffic models: L1, L2, L3, L4 (Events/Sec)	L1 (low): 10 L2 (medium): 100 L3 (heavy): 1,000 L4 (bursty): 20 (in a grid)
Transport	UDP protocol	Because of simplicity and no retransmissions
Network	CR-CEA routing method	The codes are added to NS2
Data Link	The MAC protocol presented in [33]	Channel selection is done in the network layer.
Physical	Two-ray ground reflection	Half-duplex transceivers

Table 3 Assigned values to NS2 simulation parameters

Parameter	Value
nn (Number of nodes)	226
Pt_ Sensor	0.0011 W
Pt_ Sink	0.28183 W
Pt_ ClusterHead	0.00145 W
Availability_	0.5
Simulation_Time	100 s
random-motion	0 (off)
initialEnergy_	Sensor : 1 J Cluster-Head : 2 J Sink : 4 J

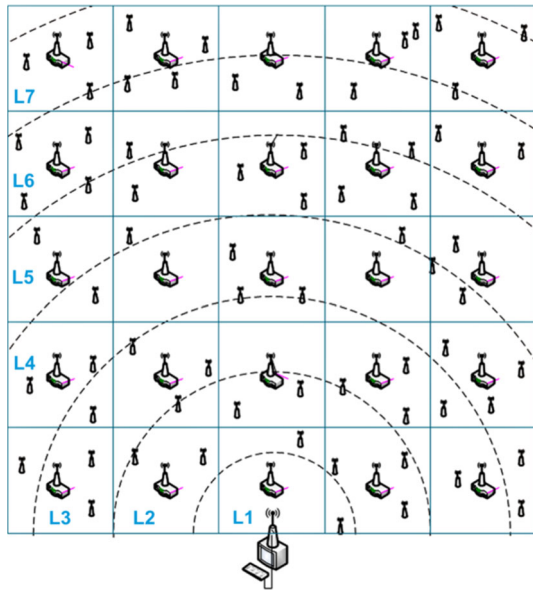


Fig. 7 The scheme of simulated CR-WSN topology

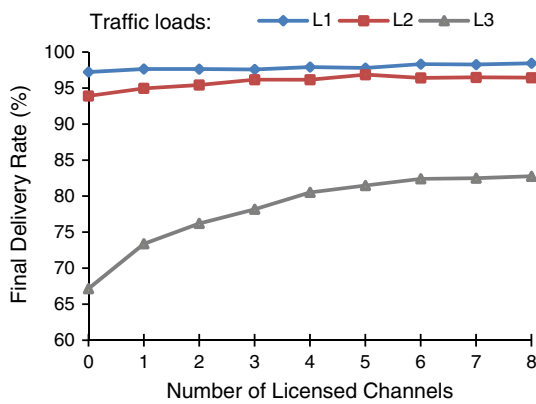


Fig. 8 Delivery rate for different number of licensed channels

similar to the situation in which low traffic *L1* flows in the network.

4.2 Energy consumption

The diagram in Fig. 10 presents the average energy of the cluster-heads in each layer. As can be observed, growth of energy consumption in cluster-heads closer to sink is greater. These nodes are always the last hops in the route, so they transmit more packets and consume more energy. Due to the fact that the width of each layer is half the r_{ch} of cluster-heads, some cluster-heads have neighbors that are located at a distance of two layers from them. This fact justifies that for example nodes in L2 are more often chosen as next-hops so have more energy consumption

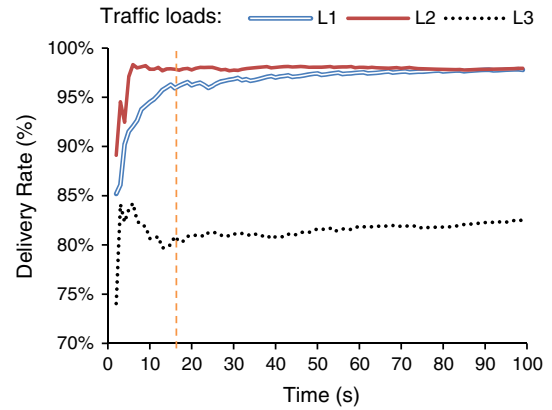


Fig. 9 Delivery Rate versus Time ($k = 5$)

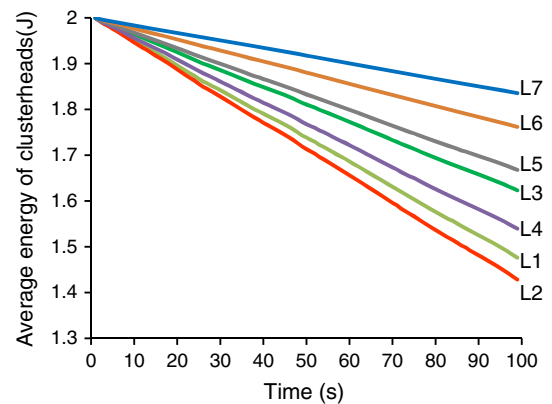


Fig. 10 Average energy of cluster heads in each layer ($k = 5$, traffic load: L3)

than nodes in L1. The uniformly generated traffic load is the cause of linear power consumption in the nodes.

4.3 Collision rate

In Fig. 11 the instantaneous collision rates are displayed for different number of available licensed channels. This rate is the ratio of the number of collisions occurred to the total number of sent messages. Obviously, when only R_0 is available in the network the collision rate is higher.

As it is mentioned at the beginning of this section, in this simulation we did not consider the reliability at the transport layer level. As a result, no packet that is dropped because of collision will be retransmitted by the transport layer of a node. Thus, the results of this simulation do not show the energy consumption, which is caused by these retransmissions. Hence it can be concluded that with an efficient decision about the working frequency, in addition to improvement in network throughput and quality of communication, we can save the energy in nodes.

4.4 Load balancing in bursty traffic

The present study was designed to determine the effect of bursty traffic load on the routing method decisions, which have a direct impact on load balancing in the network. To do so, we applied L4 traffic model to a grid in the network and perform this simulation test for all grids separately. During this simulation we monitored all selected routes from the cluster-head to the sink. Figure 12(a) demonstrates the results in a typical grid. As it is shown, the packets are passed through two different routes to the sink.

If under bursty traffic load in the network, the selected routes are mostly limited to a specific number of hops, the nodes in these routes will abruptly lose a great amount of energy. Accordingly in these situations the next hop selections should be done in such a way that the packets be forwarded through different cluster-heads. In other words, we should increase the number of possible routes in order to improve load balancing and make energy consumption

in cluster-heads more uniform, which will increase the network lifetime. With this objective we applied another condition in the next hop selection process, which increases the effect of the remainder energy of cluster-heads in this process. For this purpose, in addition to minimum W (Eq. 7), *the remainder energy of the next hop candidate should be greater than the average of all neighbors' energy*. Figure 12b shows the new results for the same grid. As can be observed, by this method of next hop selection, more routes are selected for message passing. The total results of this study indicate that by applying this condition, on average we could improve the number of routes from 2 to 5 in the considered network. This results in better load balancing.

It is worth noting that new routes may have greater hop counts. Hence, according to network requirements a tradeoff between delay and energy consumption should be made.

4.5 Layer-cost and hop count

A number of simulations have been performed with traffic loads L1, L2 and L3 and different values of *Defined_Threshold* in Cost_Calculation algorithm (20 %, 10 % and default value = 5 %). The values of Layer-Cost parameter for all layers were investigated in each simulation. The results show that, for a fixed value of *Defined_Threshold*, the values of Layer-Costs reach to their final value in the heavy traffic L3 faster than L1 and L2. This is caused by the fact that in L3 the sink receives a greater number of packets in a shorter time than two other traffics, which means the network earlier switches to the steady state. On the other hand, for the same traffic load, the values of Layer-Costs reach faster to their final value as the value of *Defined_Threshold* is greater. In these simulations, we have also captured the number of hops the

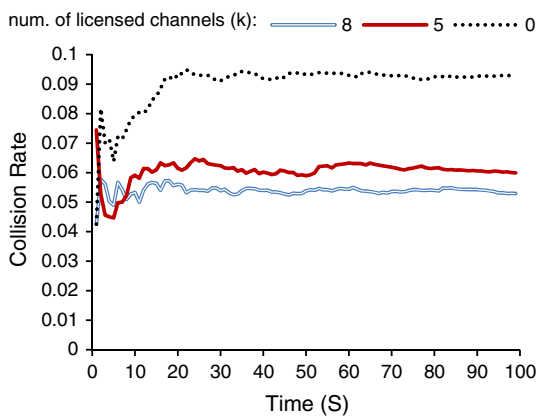
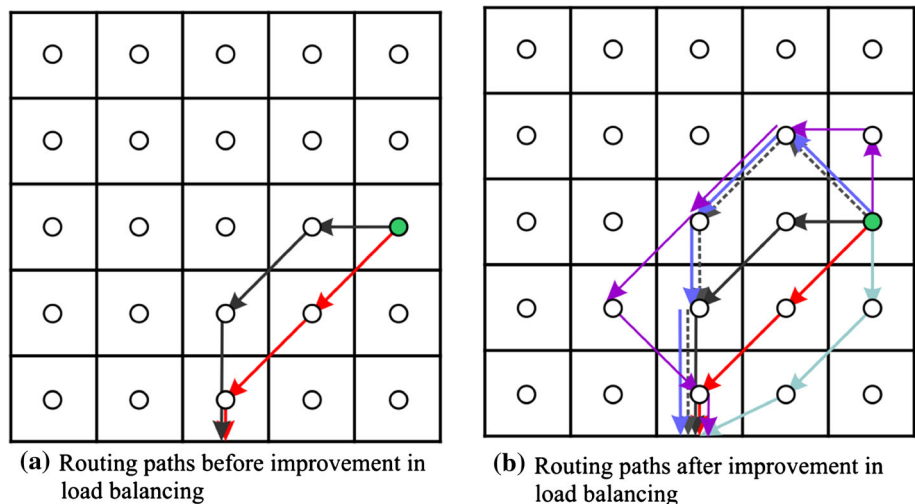


Fig. 11 Collision rate for different number of licensed channels (Traffic load: L3)

Fig. 12 Routing paths when bursty traffic (L4) flows in the network are shown. By increasing the effect of remainder energy of nodes in the next hop selection algorithm, numbers of possible routes have been increased. ($k = 5$)



packets have traversed. On average each packet had approximately experienced a delay of 2.9 hops in the steady state of the network.

4.6 Overhead

Computational overhead, messages of signaling protocols and all control messages are types of overhead that may be caused by any routing method. The overhead of CR-CEA method, which is defined as the ratio of all control messages to data messages in the network, is presented in the diagram in Fig. 13. This diagram shows the results for two different traffic loads. According to the chart it can be concluded that in this method the number of control messages is almost constant and are independent from type of traffic load.

It is worth noting that in the early stage of simulation, we have a higher level of overhead. A reasonable explanation for this is that before the time that costs of layers reach to their final constant value, in addition to PT_ChCtl messages, PT_SCtl messages are sent by sink to update the Layer-Costs. After that time, the only control messages in the network are PT_ChCtl messages which are always sent by cluster-heads periodically.

4.7 Spectrums with different qualities

In Sect. 3 it was mentioned that this method has two strategies for spectrum selection: first just presence of PU and spectrum availability with the goal of reducing the number of collisions is considered, second the quality of available spectrum is also taken into account. Up to here, the first strategy was used in all simulations. This means the channel selection process was mainly based on channel availability, PU activity, and the congestion of each channel around a specific node. Here a new situation of the network is simulated and evaluated, in which the licensed

channels are assumed to have different priorities based on their quality. The quality of channels (propagation media) depends on the environment, in which the nodes of the network are deployed.

In this situation the *channel selection algorithms* will change as follows:

- *Cognition cycle channel selection:* After cluster-head senses the spectrums, among $k-n$ available channels, it selects the one with minimum Q using Eq. 8:

$$Q(C_i) = \frac{\text{Num. of active neighbours in } C_i}{\text{Num. of neighbours}} \times \text{Priority}(C_i),$$

$$i = \{1, \dots, k\}, \tag{8}$$

where C_i is the i th licensed channel. This channel is selected as Current-Channel and is used by all sensor nodes in this cluster to communicate with the cluster-head. As they are notified by the cluster-head, they reconfigure their transceiver to work in this frequency band.

- *Routing channel selection:* before sending the packets to the next hop, the source cluster-head checks the channels around next hop in terms of *priority* and active neighbors in each channel. It calculates the value of Q according to number of next hop's neighbors and the number of active neighbors in each $C_i (i = 1..k)$. Finally, among $k-n$ available channels, the source cluster-head selects the channel, which has the minimum Q and sets C_{nx} as it. It should be said that while at least one of C_1, C_2, \dots, C_k is available, R_0 is not returned as a result of these two algorithms.

To simulate channels with different qualities, it is assumed we have a network with six channels with different loss rates. Channel R_0 has a loss rate of 20 % and $C_1,$

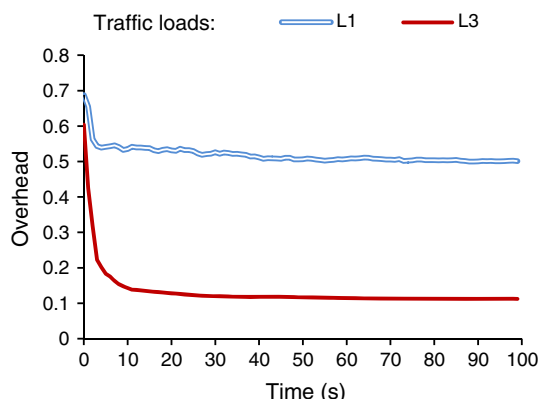


Fig. 13 Overhead - the ratio of control messages to data messages. (k = 5)

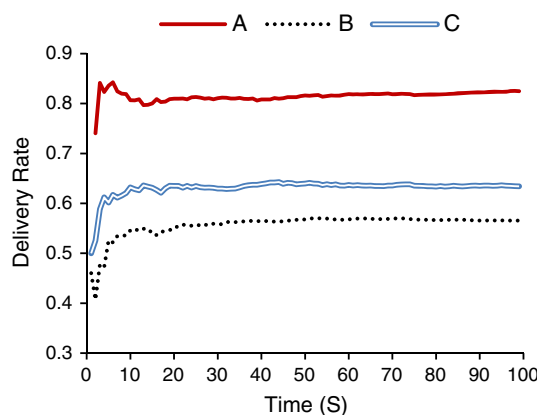


Fig. 14 Delivery rate in three network channels situations: A-channels are similar; B-channels have different loss rates but equal priorities; C-channels have different loss rates and subsequently different priorities (k = 5, traffic load: L3)

C_2, \dots and C_5 respectively have loss rates of 3, 6, ... and 15 %. Therefore, in cluster-heads channels' priorities are set to their loss rate value. It means a channel that has less loss rate has more chance to be selected. Figure 14 illustrates the delivery rate of the network in three situations. In situation A we have similar channel, in situation B we have channels with different loss rate with the same priorities and at last in situation C we have channels worth different loss rate and priorities. As it can be seen, by selecting channels based on Eq 8, an improvement on delivery rate was achieved.

In Fig. 15 the instantaneous delivery rate in the recent described situation is shown. This rate clearly varies over time as it is illustrated. PU activity that means dynamism of channel availability is the cause of this variation. As an instance, at time A of simulation, all five considered channels were available. Thus, there was an efficient distribution in channel assignments and the network has maximum delivery rate. On the other hand at time B, among all channels only C_5 with a loss rate of 15 % was available. Therefore, during this time all nodes in the network were working in this channel. In such a condition, with an increase in the number of collisions we have a minimum delivery rate at time B.

4.8 Comparison with ECR and AODV

This section includes the results of simulations, which have been performed to compare CR-CEA method with ECR algorithm [27] and AODV [23]. Two performance evaluation criteria are used: network lifetime and delivery rate. The diagram in Fig. 16 shows the number of alive nodes during the 1,000 s simulation time for these three Methods. A traffic load of 160 messages per second is applied to the network. Also the initial energy of cluster-heads is set to five joule. It is assumed that licensed channels are available 75 % of the time; this means during this test $Availability_ = 0.75$. It can

be seen that in comparison to ECR and AODV, when CR-CEA is used as routing method in this CR-WSN, all the cluster-heads in the network are alive for near 320 s. This time is longer than the other methods, which for ECR is near 50 s and for AODV is near 130 s. On the other hand, at the end of the test, the network on which ECR was running has two more alive cluster-heads than CR-CEA. Due to the definition, WSN lifetime in a clustered network, is the time until the first cluster head is drained of its energy [34]. according to these results CR-CEA made an improvement on network lifetime.

The diagram of delivery rates for these methods is presented in Fig. 17. The apparent effect of network alive cluster-heads can be clearly observed in this diagram. According to simulation results, at time 800 s equal numbers of cluster-heads are alive in both ECR and CR-CEA. From now on, ECR can not make any route setup in the network, so cluster-heads can not route any routing packets. This fact justifies the constant value of delivery rate after

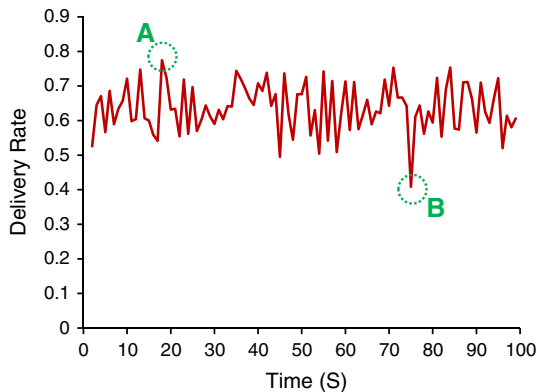


Fig. 15 Instantaneous delivery rate in spectrums with different loss rates ($k = 5$, traffic load: L3)

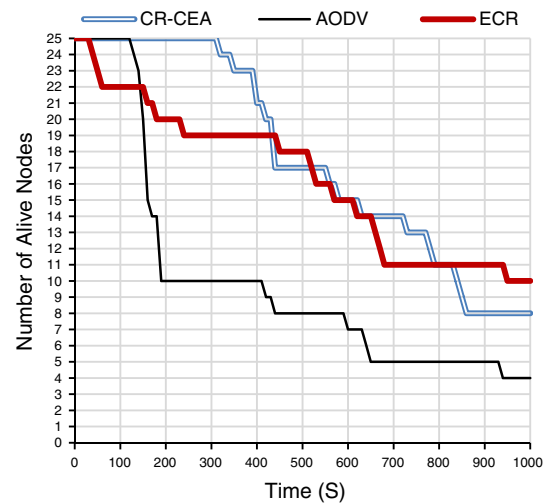


Fig. 16 Number of alive nodes (the network lifetime)

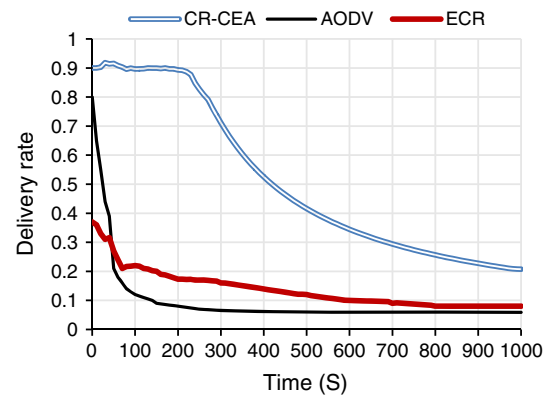


Fig. 17 Comparison of delivery rates

800th second in Fig. 17. In addition, cluster-heads lose no energy because of routing packet transmission. But in CR-CEA, routing is proceeded hop by hop by cluster-heads. So, they send more routing packet and subsequently consume more energy. The described facts justify less number of alive node for CR-CEA at the end of simulation (Fig. 16).

5 Conclusion

In this paper, we proposed the CR-CEA routing method that targets the reduction of collision occurrence probability to prevent the waste of energy in sensor nodes caused by retransmissions. To do so, it uniformly distributes frequency channels between adjacent nodes and locally decreases the collisions. The routing is performed hop-by-hop with local information and decisions. The cross-layer design of the method provides the ability to select a proper working channel in the network layer. PU activity and prevention from interference with them, is considered in all spectrum decisions, therefore it is adaptable to the time varying nature of spectrum bands. To improve the QoS parameters, in CR-CEA, route selection is also energy-aware and a learning-based technique is used to reduce the packet delay in terms of hop-count.

References

- Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., & Cayirci, E. (2002). Wireless sensor networks: A survey. *Computer Networks*, 38(4), 393–422.
- Yau, K. L., Komisarczuk, P., & Teal, P. (2009). Cognitive radio-based wireless sensor networks: Conceptual design and open issues. In *IEEE 34th conference on local computer networks, (LCN 2009)* (pp. 955–962).
- F.C. Commission. (November 2002). Spectrum policy task force, Technical report.
- Mitola, I. J., & Maguire, J. G. Q. (1999). Cognitive radio: Making software radios more personal. *IEEE Personal Communications Magazine*, 6(4), 13–18.
- FCC. (December 2003). ET Docket No 03-222 Notice of proposed rulemaking and order.
- Akyildiz, I. F., Lee, W.-Y., Vuran, M. C., & Mohanty, S. (2008). A survey on spectrum management in cognitive radio networks. *IEEE Communications Magazine*, 46(4), 40–48.
- Akyildiz, I. F., et al. (2006). NeXt generation/dynamic spectrum access/cognitive radio wireless networks: A survey. *Computer Networks: The International Journal of Computer and Telecommunications Networking*, 50(13), 2127–2159.
- Fortuna, C., & Mohorcic, M. (2009). Trends in the development of communication networks: Cognitive networks. *Computer Networks: The International Journal of Computer and Telecommunications Networking*, 53(9), 1354–1376.
- Akan, O. B., Karli, O. B., & Ergul, O. (2009). Cognitive radio sensor networks. *IEEE Network*, 23(4), 34–40.
- Vijay, G., Bdira, E., & Ibnkahla, M. (2010). Cognition in wireless sensor networks: A perspective. *IEEE Sensors Journal*, 11(3), 582–592.
- Al-Karaki, Jamal. N., & Kamal, Ahmed. E. (2004). Routing techniques in wireless sensor networks: A survey. *IEEE wireless communication*, 11(6), 6–28.
- Singh, S. K., Singh, M. P., & Singh, D. K. (2010). Routing protocols in wireless sensor networks—A survey. *International Journal of Computer Science & Engineering Survey (IJCSES)*, 1(2), 63–83.
- Cesana, M., Cuomo, F., & Ekici, E. (2010). Routing in cognitive radio networks: Challenges and solutions. *Journal of Ad Hoc Networks*, 10(5), 803–815.
- Xin, C., Ma, L., & Shen, C. (2008). A path-centric channel assignment framework for cognitive radio wireless networks. *Mobile Networks and Applications*, 13(5), 463–476.
- Zhou, X., Lin, L., Wang, J., & Zhang, X. (2009). Cross-layer routing design in cognitive radio networks by colored multigraph model. *Wireless Personal Communications*, 49(1), 123–131.
- Wang, Q., & Zheng, H. (2006). Route and spectrum selection in dynamic spectrum networks. In *3rd IEEE consumer communications and networking conference CCNC 2006*. doi:10.1109/CCNC.2006.1593099.
- Pyo, C. W., & Hasegawa, M. (2007). Minimum weight routing based on a common link control radio for cognitive wireless ad hoc networks. In *IWCMC'07: Proceedings of the 2007 international conference on wireless communications and mobile computing* (pp. 399–404).
- Xie, M., Zhang, W., & Wong, K. (2010). A geometric approach to improve spectrum efficiency for cognitive relay networks. *IEEE Transactions on Wireless Communications*, 9(1), 268–281.
- Ma, H., Zheng, L., Ma, X., & Luo, Y. (2008). Spectrum aware routing for multi-hop cognitive radio networks with a single transceiver. In *3rd international conference on cognitive radio oriented wireless networks and communications crowncom* (pp. 1–6).
- Cheng, G., Liu, W., Li, Y., & Cheng, W. (2007). Spectrum aware on-demand routing in cognitive radio networks. In *2nd IEEE international symposium on new frontiers in dynamic spectrum access networks, DySPAN 2007* (pp. 571–574).
- Cheng, G., Liu, W., Li, Y. & Cheng, W. (2007). Joint on-demand routing and spectrum assignment in cognitive radio networks. In *IEEE international conference on communications, ICC'07* (pp. 6499–6503).
- Yang, Z., et al. (2008). Local coordination based routing and spectrum assignment in multi-hop cognitive radio networks. *Mobile Networks and Applications*, 13(1), 67–81.
- Perkins, C., Belding-Royer, E., & Das, S. (2003). *Ad hoc on-demand distance vector (AODV) routing*. USA: RFC Editor.
- Pefkianakis, I., Wong, S., & Lu, S. (2008). SAMER: Spectrum aware mesh routing in cognitive radio networks. In *3rd IEEE symposium on new frontiers in dynamic spectrum access networks* (pp. 1–5).
- Chowdhury, K. R., & Akyildiz, I. F. (2011). CRP: A routing protocol for cognitive radio ad hoc networks. *IEEE Journal on Selected Areas in Communications*, 29(4), 794–804.
- Lakshmi Phani, G., & et al. (2010). ERFLA: Energy efficient combined routing, fusion, localization algorithm in cognitive WSN. In *7th international conference on wireless and optical communication networks (WOCN 2010)* (pp. 1–5).
- Oey, C. H. W., Christian, I., & Moh, S. (2012). Energy-and cognitive-radio-aware routing in cognitive radio sensor networks. *International Journal of Distributed Sensor Networks*,. doi:10.1155/2012/636723.
- Cavalcanti, D., Das, S., Wang, J., & Challapali, K. (2008). Cognitive radio based wireless sensor networks. In *Proceedings*

of the 17th International Conference on Computer Communications and Networks (ICCCN'08) (pp. 491–496).

29. Rajagopalan, R., & Varshney, P. K. (2006). Data-aggregation techniques in sensor networks: A survey. *IEEE Communications Surveys & Tutorials*, 8(4), 48–63.
30. Pohl, I. (1970). Heuristic search viewed as path finding in a graph. *Artificial Intelligence*, 1, 193–204.
31. The Network Simulator (ns), version 2. <http://www.isi.edu/nsnam/ns/>.
32. Calvo, R. A., & Campo, J. P. (2007). Adding multiple interface support in NS-2. <http://personales.unican.es/aguerocr/files/ucMultiInterfacesSupport.pdf>.
33. Zhong, J., & Li, J. (2009). Cognitive radio cognitive network simulator. Michigan Tech University. <http://stuweb.ee.mtu.edu/~ljalian/>.
34. Dietrich, I., & Dressier, F. (2009). On the lifetime of wireless sensor networks. *ACM Transactions on Sensor Networks*, 5(1), 1–38.



Rojin Tizvar received the B.S. and M.S. degrees in Computer Engineering from Shahid Beheshti University, Tehran, Iran, in 2009 and 2012 respectively. She is working as a researcher at the Wireless Networking Laboratory of Shahid Beheshti University. Her main fields of interest include Wireless Sensor Networks, Ad Hoc Networks and Cognitive Radio Networks. She is currently researching on WSN applications in smart buildings and E-Health projects.



networking.

Maghsoud Abbaspour received the B.Sc., M.S., and Ph.D. degrees in electrical engineering from the University of Tehran, Tehran, Iran, in 1992, 1995 and 2003, respectively. He has been an assistant professor of Computer Engineering department and the manager of Wireless Networking Laboratory in Shahid Beheshti University, Tehran, Iran since 2005. He is interested in Wireless Sensor Networks, peer to peer and ad hoc networks, network security and multimedia



Distributed Search Engine optimization.

Mahdi Dehghani received the B.S. degree in Computer Engineering from Shahid Beheshti University, Tehran, Iran, in 2010. Currently he is pursuing his M.S. in Computer Engineering while working as a researcher at the Web and Search Engine Laboratory of Iran University of Science and Technology. His researches are mainly concentrated on distributed data processing on Clusters and Data Grid. His current research interests also include