Distributed Protocol for Channel Assignment in Cognitive Wireless Sensor Networks

Mihaela Cardei and Amalya Mihnea
Department of Computer and Electrical Engineering and Computer Science
Florida Atlantic University
Boca Raton, FL 33431, USA
E-mail: {mcardei, amihnea}@fau.edu

***Abstract*—Cognitive radios allow secondary users to use the underutilized spectrum. However, in the presence of a primary user, the unlicensed users must vacate the spectrum, leading to a decrease in network performance or even network partition. In this paper we address the problem of robust topology control in wireless sensor networks with the objective of assigning a sensor channel on each radio such that the resulting topology is robust to the presence of a primary user. That means that if a channel is reclaimed by a primary user, the resulting secondary user topology still preserves the connectivity between any two nodes. In this paper we propose a distributed algorithm for channel assignment which has low overhead and is scalable with the number of sensor nodes. We analyze the performance of our algorithm using ns-3 simulations.**

Keywords: wireless sensor network, channel assignment, distributed algorithm.

I. INTRODUCTION

Due to the recent growth of wireless applications, the communication on the unlicensed spectrum (e.g. ISM) has become congested, while the utilization of the licensed spectrum varies between 15% and 85% temporally and geographically [6]. Cognitive radio networks is a promising solution used to address the issue of inefficient spectrum usage.

A cognitive radio is designed to operate on a wide spectrum range and can switch to a different frequency band with limited delay. This technology allows primary users (PUs) to share the spectrum with secondary users (SUs), where SUs communicate through un-assigned spectrum bands without disrupting the regular usage of the PUs. Cognitive radio networks allow SUs to take advantage of unoccupied spectrum in an opportunistic manner using dynamic spectrum access strategies.

To avoid interference with a PU, an SU must vacate the spectrum when the channel is being used by a PU. This affects ongoing communication of the SUs. The challenge occurs due to the difficulty to predict when a PU will appear in a given spectrum. To use other channels, SUs have to spend a considerable amount of time for spectrum sensing and channel switching [2]. In addition, a change in an SU channel may trigger other nodes to change their channels in a ripple effect in order to maintain the desirable topology.

In this paper we address the issue of topology control in Wireless Sensor Networks (WSNs) such that to satisfy the *robustness constraint* in the presence of a PU. The WSN is using a convergecast communication model, where data is collected from the sensors to the sink. If two sensors u and v communicate on a channel that is reclaimed by a PU, then the packet is re-routed from u to v through another radio of u. Thus packet dropping and significant delay can be avoided. There are a number of related works on channel assignment in wireless networks. Our work is different than these approaches by addressing the robust topology control issue in WSNs and by proposing a distributed approach which has low overhead and is scalable to the number of sensors, properties which are relevant to a WSN environment. Compared to the algorithm in [3], our algorithm has a better performance, as illustrated in the section V.

The remainder of the paper is organized as follows. Section II presents related works on channel assignment in wireless networks. In section III we formally define the channel assignment for a robust topology control problem and in section IV we present our distributed channel assignment protocol. Section V presents simulation results using ns-3 network simulator and section VI concludes the paper.

II. RELATED WORK

In [1] the authors introduce a centralized channel assignment algorithm, MCCA (Maxflow-based Centralized Channel Assignment), developed for multi-radio wireless mesh networks in order to maximize network capacity and reduce interference. The assignment is independent of any particular traffic profile and is done such that the most critical links (e.g. those carrying large flows) experience the least possible interference.

This paper does not address the issue of channel switching in the presence of a PU. Also the centralized mechanism proposed here is not scalable for a large network such as a WSN.

Another centralized channel assignment algorithm called UBCA (Utility Based Channel Assignment) is presented in [12]. This is a traffic independent algorithm in which the delivery probability of the wireless links and their usefulness are taken into account to make a better decision for assigning good channels to good links. This channel assignment algorithm assigns channels starting from links with higher utility, where the utility of a link is defined as the number of times that a link e participates in constructing the shortest paths between the gateway and other nodes.

UBCA is compared with three relevant channel assignment algorithms: the Common Channel Assignment (CCA) [4], the Connected Low Interference Channel Assignment algorithm (CLICA) [10], and the distributed channel assignment (ROMA) [5]. UBCA achieves significant improvement in terms of reducing the interference and increasing the network capacity.

CCA applies the same channel assignment pattern for all nodes, i.e. the first radio of all nodes is tuned to the first channel, the second radio is tuned to the second channel etc. The number of channels is therefore equal to the number of radios. CLICA is a centralized algorithm that finds connected and low interference topologies by visiting nodes in the order of their priority, which depends on their distance to a reference node (the gateway) and the number of free radios they have. ROMA is a distributed algorithm that can be used in a network with at least one gateway. Each gateway produces a channel sequence $(c_1, c_2, \ldots, c_n)$ and broadcasts it. The node which is i hops away from the gateway will select channels c_{i-1} and c_i. At the end, each node will have a common channel with its previous node on the path to the gateway, and a common channel with its neighbors at the same and lower level.

The algorithm in [14] is a static and traffic independent channel assignment algorithm that tries to minimize the overall network interference by using Tabu search. This is the first work that establishes good lower bounds on the optimal network interference.

In [11], the authors proposed a semi-dynamic and distributed channel assignment mechanism called SICA that uses game theory and takes the co-channel interference into account. It uses an online learning method to assign the best channel to each radio using information gathered during the channel sensing periods. The nodes continuously refine their decision based on changes in the wireless environment. SICA outperforms Urban-X [9], another interference-aware channel assignment mechanism, even using fewer radio interfaces per node (2 instead of 3).

Urban-X assigns channels giving priority to nodes based on the number of active flows they have: nodes having higher priority have more chances to occupy the best channels. Nodes broadcast control messages over a common channel up to two-hops neighbors. Unlike Urban-X and many other channel assignment algorithms, SICA does not use a common channel between all nodes but the synchronization is achieved through exchanging messages. The use of a common channel can be wasteful when only a few interfaces are available.

In [16] network robustness and channel interference are jointly considered when developing centralized and distributed algorithms. The proposed solutions outperform existing interference-aware approaches when primary users appear, and achieve similar performance at other times. The algorithms are compared with INSTC [15]. The problem that we address is similar to the one presented in this article. We are focusing on distributed algorithms which are applicable to large scale WSNs. The distributed algorithm presented in [16] requires multiple negotiations between nodes and may require cascaded switching of multiple users.

Another paper addressing the robust topology control in cognitive WSN is [3]. Since the performance of this algorithm is compared to our proposed solution in the simulations section V, we are describing it in more detail in the section IV-A.

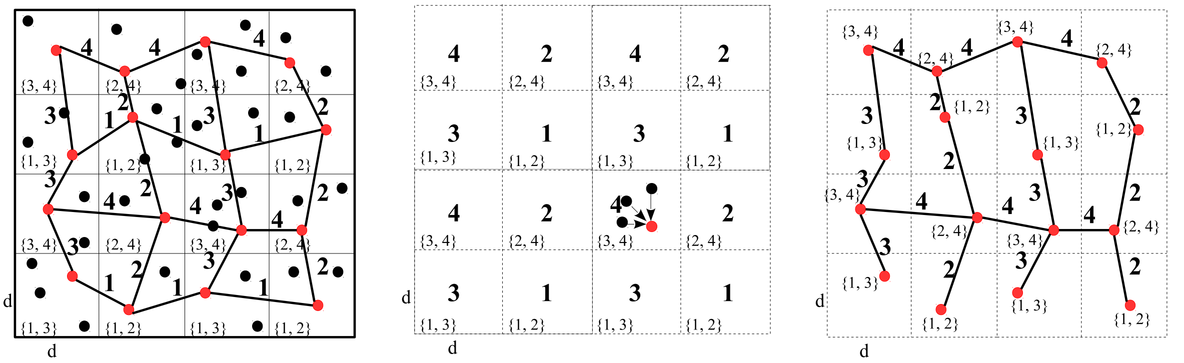
III. PROBLEM DEFINITION

In this paper we consider a WSN consisting of n homogeneous sensor nodes $s_1, s_2, ..., s_n$ and a sink node S. We assume the nodes are densely deployed and the WSN is connected. The sink node S is used to collect data and is connected to the network of sensors. Data collection follows a convergecast communication model, where data flow from many nodes (e.g. the sensors) to one (the sink).

We model the network as an undirected graph $G = (V, E)$, with the set of vertices (or nodes) being the set of sensors and the sink. An edge exists between two nodes if they are within each other's communication range.

We assume that each sensor node is equipped with Q radios and there are C channels available, where $C \geq Q$. The objective is to find a channel assignment A which assigns to each node radio a channel such that the resulting topology is connected, robust to a primary user, and has a reduced interference.

Let $A(u)$ denote the set of channels assigned to the node u, where $|A(u)| = Q$. Based on the channels assigned to the radios at each node, a channel assignment A generates a new undirected graph $G_A(V, E_A)$ where $E_A = \{(u, v, c) : (u, v) \in E \text{ and } c \in A(u) \cap A(v)\}$. Note that multiple edges may exist between two nodes if they share more than one channel, where one edge corresponds to a channel.



a. Channel allocation of each grid cell. Each cell has a *representative*, colored red.

b. Intra-cell communication.

c. Channel assignment when channel 1 is used by the primary user.

Fig. 1. Grid-based Channel Assignment

The *robustness constraint* requires that G_A is not partitioned in the presence of a primary user which communicates on a channel c_p. In that case, all the edges in G_A assigned to c_p are removed. The resulting graph must be connected. In this paper we assume that the primary user can affect part of the network or the entire network (e.g. transmission of the TV tower), but only one channel is used by the primary user at one time. We assume that the primary user is using the reclaimed channel for some amount of time.

It is easy to observe that if each node has at least two radios, then there exists a channel assignment that satisfies the robustness constraint. Just consider the case when all nodes have assigned the two channels c_1 and c_2 to both radios. Then if the primary user uses one of the channels, let's say channel c_1, then the topology remains connected using the channel c_2. The drawback of such an assignment is a high interference.

Transmissions on different channels can run in parallel with no interference. In order to reduce the network interference, the objective is to assign communication on nearby edges on different channels. If edges within interference range have assigned the same channel, then interference has to be addressed at the MAC level.

Channel Assignment for a Robust Topology Control (CA-RTC) Problem: Given a graph G find a channel assignment A such that $G_A = \{(u, v, c) : (u, v, c) \in E_A\}$ is robustly connected for any channel c and the network interference is minimized.

According to the proof in [16], the CA-RTC problem is NP-complete. We consider that the WSN is homogeneous and all the sensor nodes have the same transmission range and the same interference range. The MAC protocol used is IEEE 802.11 DCF [7]. Each link is supporting communication in one direction at one time (half-duplex).

The main contribution of this paper is the distributed channel assignment protocol described in the section IV-B.

IV. SOLUTIONS FOR THE CA-RTC PROBLEM

A. *Grid-based Channel Assignment*

The Grid-based Channel Assignment proposed in [3] is described briefly in this section. The monitored area is divided into grids, see Figure 1. Let r be the communication range of each sensor. It is assumed that sensors know their location information using GPS or other localization protocols [8]. In addition, since sensor wireless networks are densely deployed, it is assumed that each grid cell has at least one sensor.

The *neighboring cells* of a certain cell are those placed above, below, left, and right. The grid size is $d = r/\sqrt{5}$, see Figure 1, so that any two sensors in neighboring cells can communicate directly.

Consider the case when each sensor node has $Q = 2$ radios and there are $C = 4$ channels available. Each sensor computes the grid cell that it belongs to based on its GPS coordinates. A static channel assignment can be allocated in this case, as illustrated in the Figure 1.

Figure 1a. shows the channels used to communicate between neighboring cells, while Figure 1b. shows the channels allocated for the communication inside a cell. For example, the representative of the bottom leftmost cell is assigning the channels $\{1, 3\}$ to its radios, it uses channel 1 to communicate with the right representative and channel 3 to communicate with the representatives placed above. Also, the representative of this cell uses channel 3 for the intra-cell communication.

Communication between cells is accomplished through cell *representatives*. Each cell locally selects a representative, which can be for example the sensor node with the largest remaining energy. In case of a tie, the representative role is assumed by the sensor node with the largest ID.

The cell representative is in charge with forwarding messages between cells and with transmission of messages from/to the nodes inside the cell. It has to be noted that the representative consumes the largest amount of energy in the cell. In order to avoid sensor energy depletion, representatives have to be re-elected periodically.

The grid-based channel assignment mechanism satisfies the robustness constraint, that is the topology remains connected when a primary user reclaims any of the channels. Figure 1c. shows the resulting topology when channel 1 is reclaimed by a primary user. The inter-cell communication topology still ensures communication between any cells.

Intra-cell communications which originally were scheduled on the channel 1 now take place on the other assigned channel. For example, in Figure 1b., grid cells with channel assignment $\{1, 2\}$ are using channel 1 for intra-cell communication. If channel 1 is reclaimed by a primary user, then intra-cell communication switches to channel 2.

The proposed mechanism has a low overhead and can be easily extensible to the case with more radios and more channels, see the discussion in [3].

B. Distributed Channel Assignment

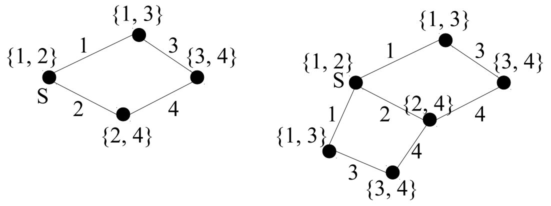
In this section we propose a distributed protocol for channel assignment. We assume that the sensor network is connected to the sink when nodes have communication range r. This mechanism has two phases:

- Phase 1: neighbor discovery and setting up the distance from the sink
- Phase 2: channel assignment

Below, we describe the mechanism for $Q = 2$ radios and $C = 4$ channels, and then we explain how it can be modified for a larger number of radios and/or channels.

1) Neighbor Discovery and Setting Up the Distance from the Sink: In this phase, all nodes use the same channel, let us say channel 1, to communicate. We assume that each node has a unique ID. Each node broadcasts a *Hello* message containing the node ID. To reduce the probability of interference, a node waits a random delay before sending the broadcast. The neighbor information is stored locally by each node. The sink node participates in this step as well.

The nodes that have the sink as a neighbor resend the *Hello* message containing their neighbor table. In this way, the sink collects information on its 2-hop neighborhood, while all other sensor nodes have 1-hop neighbor information.



a. Channel assignment for nodes on a cycle of length 4

b. Channel assignment if some of the nodes have been assigned channels

Fig. 2. Channel Assignment for Cycles of Length 4

In the second part of this phase, the sink broadcasts a message *Hops*, which contains a parameter $hops$ - the number of hops from the sink. A node receiving a *Hops* message will retransmit the message in two cases: (i) if this is the first *Hops* message received, or (ii) if this message has a shorter distance to the sink. In both cases the node updates its shortest distance to the sink, increments the $hops$ counter, and then retransmits the *Hops* message.

At the end of this phase, the sink has 2-hop neighborhood information, and each node knows its number of hops from the sink.

2) Channel Assignment: In this protocol, the nodes assign channels starting from the sink, in an incremental approach. The sink S has two radios, similar to all other nodes. S chooses two channels arbitrarily for its radios, let us say channels 1 and 2. Next, the sink S uses the 2-hop neighborhood information acquired in the previous step to assign channels to the sensor nodes on cycles of length 4 incident to S. The mechanism is shown in *SinkLocalNeighborhoodAssignChannels*(S) procedure where S is the sink.

The sink assigns channels to cycles of length 4. The objective is that each pair of neighboring nodes communicate on a different channel, so that all four channels are used. The edges incident to the sink will have values 1 and 2 in arbitrary order. The two other edges on the cycle are assigned values 3 and 4 in arbitrary order. Figure 2a. shows an example of channel assignments for the nodes on a cycle. An alternative assignment is when nodes are being assigned values $\{1, 2\}$, $\{1, 4\}$, $\{3, 4\}$, and $\{2, 3\}$, where $\{1, 2\}$ is the assignment of the sink S. One of these two assignments is assigned to each cycle, arbitrarily.

If some of the nodes on a cycle have been already assigned channels, then the others are assigned channels following the same rule (e.g. each of the four channels on an edge), see Figure 2b.

The main goal in assigning channels to cycles of length 4 is to use all channels in order to increase diversity and to reduce communication interference. Cycles of length 4 can be computed using a depth-first search

Algorithm 1 SinkLocalNeighborhoodAssignChannels(S)

```
compute all cycles of length $4$
for each cycle of length $4$ do
    assign channels to all nodes on the cycle that have not assigned their channels yet, such that all 4 channels are used on the cycle
end for
broadcast SinkLNChannelSet containing all channel assignments to its 2-hop neighborhood (TTL = 2)
wait a random time and broadcast ChannelSet(S, 1, 2, TTL = 1)
```

approach starting from S, and checking paths of length 4. The algorithm has to check adjacency lists of the first three nodes on the path, and if the fourth node is a sink neighbor. The complexity is $O(\alpha^3)$, where α is the node degree.

The sink then broadcasts a message *SinkLNChannelSet* to its 2 hop-neighborhood, containing a list with channels assigned by the sink so far (e.g. sensor nodes on cycles of length 4). Nodes in the list receiving this message will assign their channels accordingly. In order to broadcasts this message to the 2 hop-neighborhood, the message will have a TTL = 2. The sink and the nodes whose channels have been set up at this step will then broadcast a *ChannelSet* message to their 1-hop neighbors (TTL = 1) to advertise their channels. These messages are sent with a small random delay to avoid interference.

Let us consider the example in Figure 3. The sink computes three cycles: (S, G, F, D), (S, G, K, J), and (S, B, A, C) and assigns channels to the nodes $S, G, F, D, K, J, B, A, C$. A list containing these channel assignments is sent using a *SinkLNChannelSet* message with TTL = 2. All these nodes in the list will assign their channels accordingly and then broadcast a *ChannelSet* message with TTL = 1. At the end of this step, all the nodes will have their channels assigned except the nodes E, H, I.

Sensor nodes wait for *SinkLNChannelSet* and *ChannelSet* messages in order to select their channels. The mechanism used by a sensor node v is shown in the *AssignChannels*(v) procedure.

If a *SinkLNChannelSet* message is received and v has been assigned channels $\{x, y\}$ by the sink, then v uses those channels and broadcasts *ChannelSet*(v, x, y, TTL = 1) after a random time to inform its neighbors about its channels selection.

Sensor nodes without a channel assignment wait a time proportional to the distance from the sink (e.g. number of hops from the sink v_{hops}). The waiting time is computed as $Time(v_{hops}) = v_{hops} \times hopDelay$, where $hopDelay$ is the delay per hop and it must account for the propagation delay, algorithm execution time, and the maximum waiting time of a node before sending the *ChannelSet* message. In this way the nodes at distance 1 will set up their channels first, followed by the nodes at distance 2, then 3, and so on.

Algorithm 2 AssignChannels(v)

```
if $v$ receives channel assignments $\{x, y\}$ in a SinkLNChannelSet message then
    node $v$ assigns channels $\{x, y\}$ to its radios
    node $v$ waits a random time and broadcasts ChannelSet(v, x, y, TTL = 1)
    return
end if
set the waiting time $t = Time(v_{hops})$
record channels assigned by neighbor nodes based on ChannelSet messages received
when timer t expires, examine the recorded neighbor channels and compute the two least used channels $x$ and $y$
node $v$ assigns channels $\{x, y\}$ to its radios
node $v$ waits a random time and broadcasts ChannelSet(v, x, y, TTL = 1)
```

Considering the example in the Figure 3a., nodes H and E are at distance 2 from the sink and they will establish their channels first. Node H knows the channels assigned by G and K, and will select the two least used channels $\{2, 3\}$. Node E knows the channels assigned by D and F and will assign channels $\{1, 4\}$. After nodes at distance 2 have set up their channels, nodes at distance 3 will follow. In our example, the node I is at distance 3 from the sink and will select the least used channels $\{2, 4\}$.

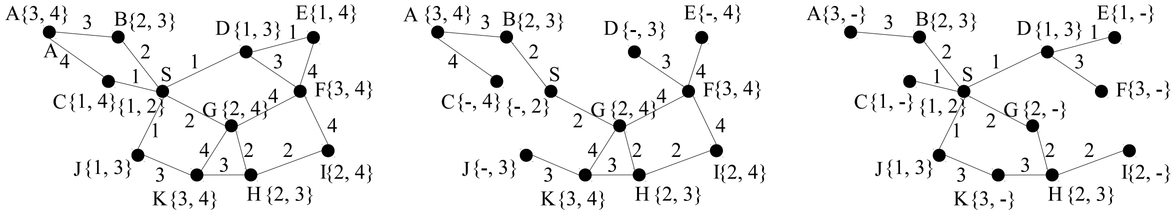
The proposed channel assignment mechanism satisfies the robustness constraint, that is the topology remains connected when a primary user reclaims any of the channels. Figures 3b. and 3c. show the resulting topology when channel 1 or 4 is reclaimed by a primary user.

Theorem Assuming that the starting wireless sensor topology is connected to the sink, the Distributed Channel Assignment algorithm terminates in finite steps and achieves robustness upon termination.

Proof: The proof is by induction. The starting point of the algorithm is the sink S, which assigns channels 1 and 2 to its radios. Let us assume that all nodes that have sent the *ChannelSet* message are robustly connected to the sink; that means that if one channel is reclaimed by a primary user then the node is still connected through a path to the sink.

Let us take a node v which has just assigned its channels using the *AssignChannels* procedure. If v has received only one message of type *ChannelSet* from a node u, then v will assign the same two channels as u, so it will be robustly connected to the sink since u has this property based on the inductive assumption.

If the node v has received two or more *ChannelSet*



a. Example of channel assignment

b. Resultant topology when channel 1 is reclaimed by the primary user

c. Resultant topology when channel 4 is reclaimed by the primary user

Fig. 3. Example using the Distributed Channel Assignment Algorithm

messages, then v selects two channels used by its neighbors. If one of the channels is reclaimed by the primary user, then v remains connected to at least one neighbor u. Since u is robustly connected to the sink based on the inductive assumption, then v will remain connected to the sink after the primary user appearance. ■

The Distributed Channel Assignment protocol has a low overhead of $O(n)$ messages, where n is the number of nodes. Each node is sending a *Hello* message, one (or few) *Hops* messages, and a *ChannelSet* message, while the nodes in the 2-hop neighborhood of the sink send two extra messages.

The method presented above for $Q = 2$ radios and $C = 4$ channels can be easily extended if there are more radios and/or channels available. The algorithm starts from the sink, which finds cycles of length at most C.

For this the sink needs $\lfloor C/2 \rfloor$-hop neighborhood information. The sink then assigns channels to the nodes on the cycles and broadcasts this information using a *SinkLNChannelSet* message. Other nodes assign their channels incrementally, similar to the mechanism described above. The main objective is to have the nodes select the least used channels, in order to reduce interference.

V. SIMULATION

In this section we evaluate the performance of our Distributed Channel Assignment (Distributed-CA) mechanism using ns-3 network simulator [13]. The performance of this algorithm is compared to the Grid-based Channel Assignment (Grid-CA) [3] mechanism.

A. Simulation Environment

In the simulations, we set the node communication range $r = 100$m. To be able to compare our protocol with the Grid-CA, we consider that the deployment area is a square divided into grids. We vary the number of rows (which is the same as the number of columns) between 5 and 25 with increment of 4, see Figure 4. The grid cell size is computed as $d = r/\sqrt{5}$. As a result the deployment area side varies between 223m and 1118m and the number of sensor nodes n varies between 75 and 1875. The sensors are deployed randomly in the monitoring area, and we place the sink S in the middle of the area.

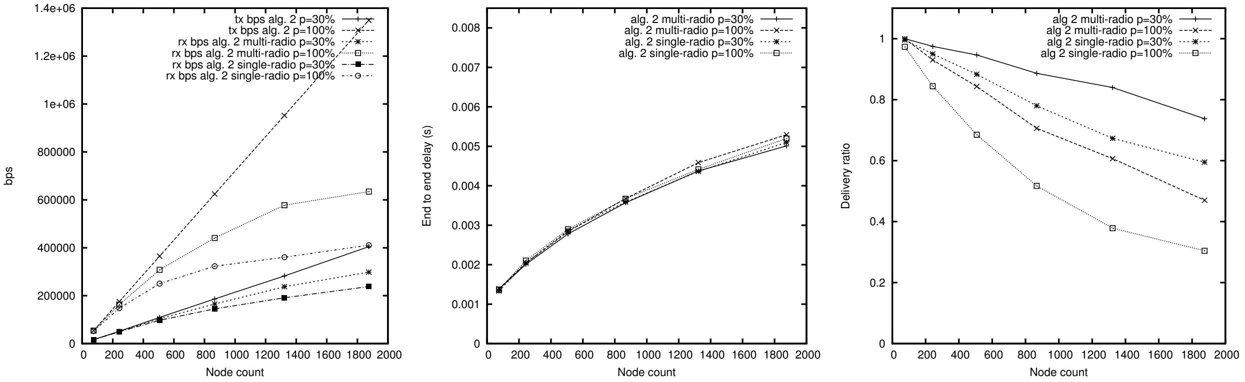
Number of rows	Number of cells	Area side length, m	Number of sensors
5	25	223.607	75
9	81	402.4926	243
13	169	581.3782	507
17	289	760.2638	867
21	441	939.1494	1323
25	625	1118.035	1875

Fig. 4. Simulation parameters

The transmission rate for the wireless radio is 1Mbps. In our simulation, we consider that the nodes have $Q = 2$ radios and $C = 4$ channels. Once the sensors are deployed, they use a channel assignment mechanism to assign channels to their radios. To test the performance of the resulting topology, we employ the following shortest-path data gathering protocol.

The sink S broadcasts a beacon message in the whole network. Each sensor node sets up its routing table with the next hop being the node from which the beacon with the smallest number of hops to the sink was received.

We use a convergecast communication model where traffic flows from the sensor nodes to the sink. At the MAC level we use CSMA for the wifi channels. Each sensor node has a parameter p - the probability that the node sends a message in each iteration (e.g. each second). Sensors send 656 bytes data packets every second. In the simulations, we represent two cases: when $p = 100\%$ and $p = 30\%$. We run each simulation scenario 5 times using different seed numbers and report the average values in the graphs. Each simulation scenario is run for 20 seconds.



(a) (b) (c)

Fig. 5. Distributed-CA: comparisons between multi-radio and single-radio WSNs (a)Data throughput. (b)End-to-end delay. (c)Delivery ratio.

B. Simulation Results

In the first experiment the sensor nodes use the Distributed-CA protocol (denoted alg. 2 in the figures) to assign channels to their radios. We compare the performance of the network when the nodes are equipped with multi-radios multi-channels, versus the case when nodes have single-radio single-channel. Simulation results are presented in Figure 5. Two cases are considered, when nodes send data packets every second with probability $p = 30\%$ and $p = 100\%$. Figure 5a. shows both data transmitted by the sensors as well as data received by the sink. We can observe that in both cases a higher throughput is received for multi-radio WSNs. Some of the data packets are lost due to collisions. It is known that collisions increase as packets get closer to the sink.

Figure 5b. compares the end-to-end delay. This metric is larger for $p = 100\%$ since more packets are being transmitted. We also observe that the single-radio network has a slightly smaller delay. This is because in the multi-radio topology the shortest path to the sink may have a larger length than in the single-radio topology.

In Figure 5c. we can observe that the delivery ratio decreases with an increase in the number of sensors, due to the collisions in the network. If nodes send with $p = 30\%$, then a higher delivery ratio is achieved due to a smaller number of collisions. This graph also illustrates the advantage of a multi-radio network, which has an increased delivery ratio compared to single-radio networks.

In the second experiment we compare the performance of the two algorithms, Grid-CA (alg 1) and Distributed-CA (alg 2), in Figures 6 and 7. In Figure 6 nodes send data with probability $p = 100\%$, while in Figure 7 nodes send data with probability $p = 30\%$. The results in these figures are consistent.

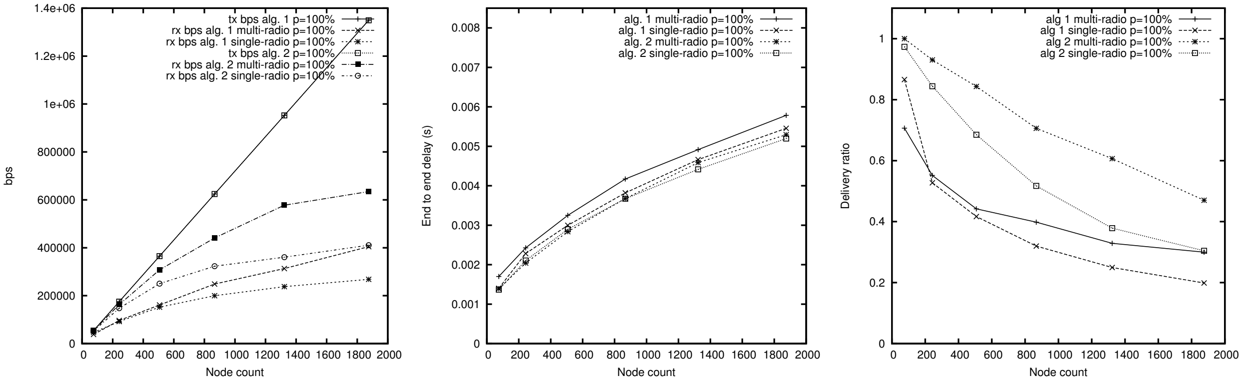
Figures 6a. and 7a. show that the Distributed-CA has a higher throughput than the Grid-CA mechanism. This is due to a smaller number of collisions, since the nodes in the same grid cell compete when sending messages to the same representative. The figures also show a higher throughput when nodes are equipped with multi-radios, illustrating the benefit of using multiple transmissions without interference on different channels.

The end-to-end delay is analyzed in Figures 6b. and 7b. Overall, the two algorithms have comparable end-to-end delays which increase with the size of the network. The Distributed-CA has a slightly smaller delay, since nodes may find a shorter path than in the case when data is forwarded through the cell representatives. The single-radio single-channel case has a slightly smaller delay than the multi-radio multi-channel one since nodes may find shorter paths to the sink.

Figures 6c. and 7c. present the delivery ratio. Distributed-CA has a higher delivery ratio due to a reduced number of collisions. Multi-radio networks have a higher delivery ratio due to channel diversity which decreases the number of collisions.

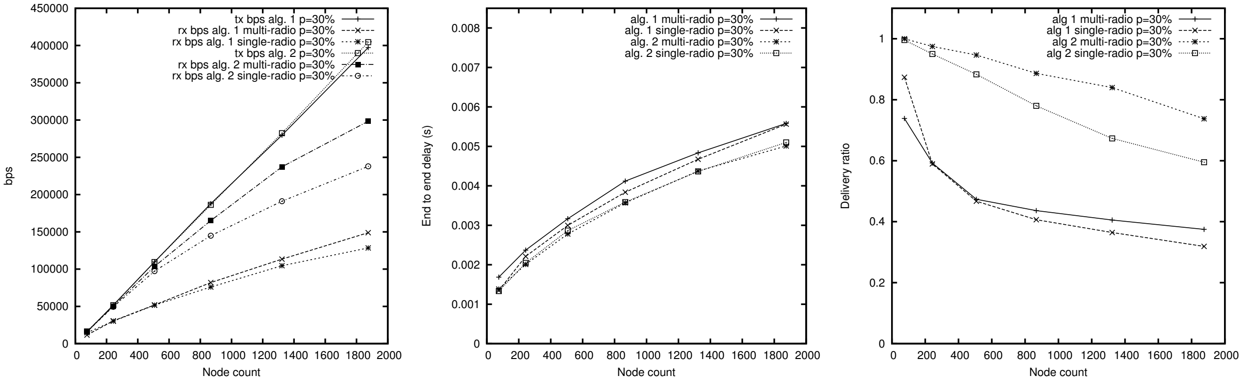
In the third experiment we test the behavior of the network in the presence of a primary user. In Figure 8 the percentage of the area affected by the primary user varies between 0 (no PU) to 1 (PU is affecting the whole area). The PU is using a single channel and two scenarios were considered depending on whether the PU channel is identical to a sink channel or not. When the PU is on the same channel as the sink, the sink will use the other radio for wireless communication.

In the presence of a PU the topology is still connected, even though it is sparser. Note that in our simulations we consider that the sink is placed in the middle of the monitored area. We take the monitored area to be a square with side L. The area affected by the PU is taken to be the rectangle with height L and width



(a) (b) (c)

Fig. 6. Comparisons between Distributed-CA (alg. 2) and Grid-CA (alg. 1). Multi-radio versus single-radio WSNs when p = 100%. (a)Data throughput. (b)End-to-end delay. (c)Delivery ratio.



(a) (b) (c)

Fig. 7. Comparisons between Distributed-CA (alg. 2) and Grid-CA (alg. 1). Multi-radio versus single-radio WSNs when p = 30%. (a)Data throughput. (b)End-to-end delay. (c)Delivery ratio.

$PU_{fraction} \times L$, starting from the origin.

In Figure 8a. we can observe that a higher drop in data rate occurs when $PU_{fraction} \geq 0.6$ and the PU is using one of the sink channels. In these cases the sink is in the area affected by the PU and it can communicate on a single radio only. This will reduce network capacity at the sink.

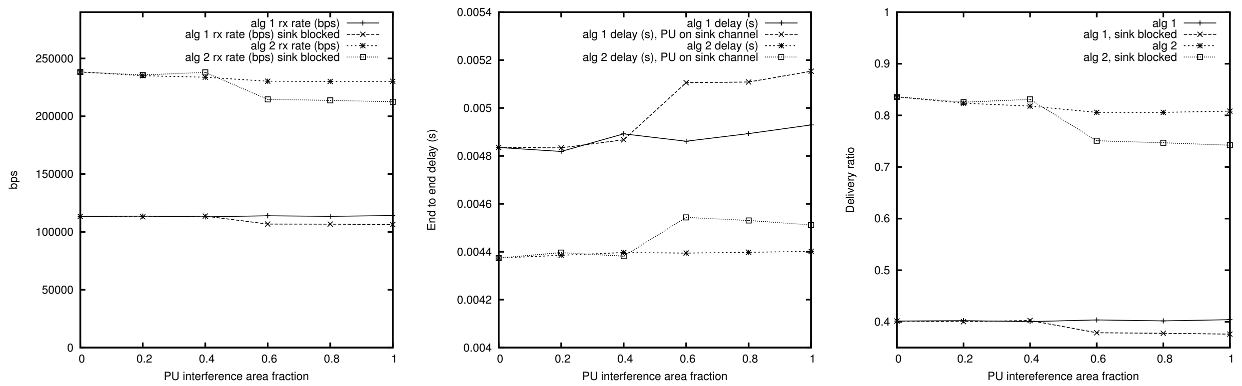
If the sink is in the area affected by the PU we can also see an increase in the end to end delay, especially in the case when the PU is on a sink channel, see Figure 8b.

Figure 8c. is consistent with the previous graphs and it shows a decrease in the delivery ratio as the sink is affected by the primary user. Even though not represented in this graph, it is evident the drastic impact of a PU on a single-radio network. In such a case, for $PU_{fraction} \geq 0.6$ the delivery ratio is 0.

In summary, the topology resulted by applying our Distributed Channel Assignment mechanism is robust to the presence of a primary user. Simulation results show the benefit of using multi-radio networks and show the network performance in the presence of a primary user. The protocol has better performance compared to the related algorithm Grid-based Channel Assignment, which requires traffic to flow through the cell representatives.

VI. CONCLUSIONS

In this paper we propose the Distributed Channel Assignment protocol which is robust to the presence of a primary user on a certain channel. In such an event, the nodes are able to continue to deliver data to the sink following the same or a different path. In the protocol, nodes assign their channels starting from the sink in an incremental approach, using channels less used by their neighbors, while maintaining the robustness constraint. Simulation results using ns-3 show the benefit of using a multi-radio topology and the robustness of the proposed Distributed Channel Assignment protocol in the presence



(a) (b) (c)

Fig. 8. Comparisons between Distributed-CA (alg. 2) and Grid-CA (alg. 1). The impact of Primary User on WSN performance. (a)Data throughput. (b)End-to-end delay. (c)Delivery ratio.

of a primary user.

ACKNOWLEDGMENT

This work was supported in part by the NSF grant IIP 0934339.

REFERENCES

[1] S. Avallone and I. F. Akyildiz, A channel assignment algorithm for multi-radio wireless mesh networks, *Proceedings of 16th International Conference on Computer Communications and Networks, (ICCCN)*, 31(7):1343-1353, 2008.

[2] P. Bahl, R. Chandra, T. Moscibroda, R. Murty, and M. Welsh, White space networking with wi-fi like connectivity, *ACM SIGCOMM*, 2009.

[3] M. Cardei and A. Mihnea, Channel assignment in cognitive wireless sensor networks, *International Conference on Computing, Networking and Communications (ICNC)* 2014, accepted.

[4] R. Draves, J. Padhye, and B. Zill, Routing in multi-radio, multi-hop wireless mesh networks, *ACM MobiCom*, 2004.

[5] A. Dhananjay, H. Zhang, J. Li, and L. Subramanian, Practical, distributed channel assignment and routing in dual-radio mesh networks, *ACM SIGCOMM*, 2009.

[6] ET Docket No 03-237 Notice of proposed rule making and order, FCC, 2003.

[7] IEEE 802.11 Working Group, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, 1997.

[8] H. Karl and A. Willig, Chapter 9: Localization and position, *Protocols and Architectures for Wireless Sensor Networks*, John Wiley & Sons, 2005.

[9] W. Kim et al., Urban-X: Towards distributed channel assignment in cognitive multi-radio mesh networks, *IFIP Wireless Days*, Venice, Italy, 2010.

[10] M. K. Marina, S. R. Das, and A. P. Subramanian, A topology control approach for utilizing multiple channels in multi-radio wireless mesh networks, *Computer Networks*, 54(2):241-256, 2010.

[11] M. A. Nezhad and L. Cerdà-Alabern, Adaptive channel assignment for wireless mesh networks using game theory, *Proceeding of the IEEE 8th Int Mobile Adhoc and Sensor Systems (MASS)*, pages 746-751, 2011.

[12] M. A. Nezhad, L. Cerdà-Alabern, and M. G. Zapata, UBCA: A centralized utility based channel assignment mechanism for multi radio mesh networks, *Scientific Research and Essays*, 2011.

[13] ns-3 network simulator, http://www.nsnam.org, 2013.

[14] A. Subramanian, H. Gupta, S. Das, and J. Cao, Minimum interference channel assignment in multiradio wireless mesh networks, *IEEE Transactions on Mobile Computing*, pages 1459-1473, 2008.

[15] J. Tang, G. Xue, and W. Zhang, Interference-aware topology control and QoS routing in multi-channel wireless mesh networks, *ACM MobiHoc*, 2005.

[16] J. Zhao and G. Cao, Robust topology control in multi-hop cognitive radio networks, *IEEE Infocom*, 2012.