This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE CCNC 2010 proceedings

Allocation of Opportunistic Spectrum in Cognitive Radio Ad hoc Networks

Vijay S Rao, R Venkatesha Prasad, Chetan Yadati, I.G.M.M. Niemegeers
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology, The Netherlands
Email: {V.SathyanarayanaRao, R.R.VenkateshaPrasad, C.Yadati, I.Niemegeers}@tudelft.nl

***Abstract*—Cognitive Radios (CRs) address the problems of spectrum scarcity and under-utilization of the spectrum. However, realizing a CR network is neither easy nor straight-forward. The link layer in CR ad hoc networks is responsible for choosing suitable channels, out of the available channels, for setting up communication between nodes. However, the selection of channels amongst the CR nodes in the network is proven to be NP-complete. The need of maximizing the spectrum efficiency in a fair way can be modeled as a graph coloring problem. An Edge coloring heuristic and a Clique determination heuristic algorithms are proposed in this article. We compare the results of these heuristics with the algorithm proposed in [1]. We also compare the spectrum utilization in all these algorithms.**

***Index Terms*—Cognitive Radio Ad hoc networks, MAC, scheduling, graph coloring, distributed heuristics**

I. INTRODUCTION

The recent licensed spectrum utilization is found to be between 15% to 85% [2] and DARPA in its survey reports an average of 6% usage [3]. Cognitive Radios (CRs) are envisioned as a solution to increase the spectrum utilization. A *CR* learns about its operating environment and adjusts its operating frequencies using Software Defined Radio (SDR), and further decides the transmission parameters such as channels, operating power, etc. When no *Primary users* (PUs) are using a certain licensed band, then that band can potentially be used by *Secondary Users* (SUs) or *CR users* without affecting the PUs. This *Opportunistic Spectrum Access* (OSA) can address the spectrum scarcity. With OSA, DARPA [3] envisages a tenfold increase in spectrum utilization. However, this is possible only if SUs have an "intelligent" link layer that provides collision-free (with respect to SUs) and interference-free (with respect to PUs) communication. This paper proposes to evolve algorithms to have such a communication in cognitive radio ad hoc networks.

In a *Cognitive Radio Ad hoc Network* (CRAN) the nodes form a network to support a specific purpose or application. To achieve the goals of the application, the SUs have to share the spectrum efficiently. The availability of spectrum is detected by a spectrum sensing functionality. Based on the location of the node and accuracy in spectrum sensing, each node may detect a different set of available channels. We note here that the cooperative spectrum sensing could be used to bring a homogeneous view of the available spectrum bands to an extent possible. However it is not guaranteed when distributed decision making is used, for reasons such as, some nodes located in shadowing area. In the absence of a central controller, a naive approach to spectrum access could result in collisions and interference since we consider the overlay model of CR communication. This results in under-utilization of the spectrum. Thus newer approaches to achieve an interference-free and collision-free spectrum allocation for CRANs, that increases the spectrum utilization, are essential. We model the CRANs using an undirected graph as shown in Fig. 1, with nodes denoting the vertices, and edges denoting that at least one channel is available between nodes. In Fig. 1, all available channels for the edges are shown within braces. Since we opportunistically access the spectrum, it is necessary to have an optimal link (or MAC) layer schedule. Graph coloring as a tool for channel allocation for wireless networks [4] [5] [6] has been well investigated; recent works [7] [1] have extended this concept to CRANs. It is well-known that an optimal edge coloring of a graph is an $\mathcal{NP}$ complete problem [8]. The heuristics proposed in earlier studies do not use the spectrum whenever it is available to increase the spectrum utilization. The approach is more towards finding a feasible schedule rather than taking into account utilization of the spectrum. In this work, we propose two algorithms, a modified Edge coloring heuristic algorithm and a Clique based heuristic algorithm to allocate the spectrum efficiently.

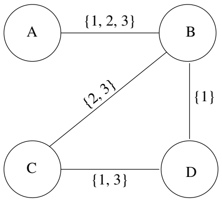

Fig. 1. Graph representation of a CRAN.

The paper is organized as follows. Section II summarizes the previous work on spectrum allocation for wireless network that use graph-theoretic models. In Section III we describe the system model and assumptions, and state the problem definition. In Section IV we propose heuristics for spectrum allocation. In Section V we evaluate the methods against previous work and we conclude in Section VI.

II. RELATED WORK

Graph coloring for channel allocation has been investigated extensively in cellular networks, where each base-station

978-1-4244-5176-0/10/$26.00 ©2010 IEEE

is to be allocated with a non-interfering channel [5] with respect to its neighboring base-stations. Edge coloring has also been proposed for link scheduling in multihop wireless networks [4] [6]. In [4], a two hop edge coloring is proposed i.e., the adjacent edges and edges which are one hop away are assigned different colors. This coloring scheme avoids hidden terminal problem but suffers from exposed terminal problem. The distributed edge coloring heuristic proposed in [6], assigns each edge a color, where color is equivalent to a timeslot. These solutions if applied for CRANs result in degraded performance of the network, since the number of channels available at each node may vary.

In [7], spectrum allocation for CRs is presented based on vertex coloring for color-sensitive graphs. A color-sensitive graph $G(U, E_C, L_B)$ is a graph with the set of vertices U, the set of colored edges E_C, and link weights L_B. The colors represent the channels and each vertex is assigned a set of channels based on a greedy heuristic. Several optimizations are proposed to optimize bandwidth or fairness in with only spectrum allocation, however scheduling is not addressed.

For CRANs, an Integer Linear Programming (ILP) based and a distributed heuristic based solutions are proposed in [1]. The ILP based approach has high computational complexity and does not scale well. The distributed heuristic proposed involves two phases. In phase 1, a $(timeslot, channel)$ pair is assigned to every link, and in phase 2, the schedule length is propagated through the network. In phase 1, every node is ranked based on its degree or number of channels available, or both. A node with the highest rank in its two hop neighborhood for which $(timeslot, channel)$ has not been assigned yet, starts the assignment operation. Once all its edges are assigned $(timeslot, channel)$ pairs, the node is said to be covered and this assignment is distributed amongst its two hop neighborhood. Each time an arbitrary node receives a schedule it updates its knowledge of the current schedule, and it checks if it is the highest ranking uncovered node in its two-hop neighborhood to perform assignment operation. In this distributed greedy heuristic, assignment procedure is not executed in parallel at all the nodes, and each node waits for its turn until it becomes the highest ranked uncovered node to cover its edges in its two hop neighborhood. This ensures a conflict-free assignment but requires more time. In the process of conflict-free assignment, the schedule length deviates from the optimal. This delay may result in wastage of spectrum and there is a higher possibility that the set of available channels might change.

While it is necessary to get a conflict-free schedule, in our approach we do not avoid the conflicts but resolve them. This results in overall lower schedule length. Thus we address this issue in this article.

III. SYSTEM MODEL AND PROBLEM DEFINITION

The CR nodes have to coexist, and the use of spectrum opportunistically. The spectrum allocation problem is to find an optimal collision-free schedule in which the nodes that have data to transmit to its neighbors get a time-slot and a channel.

A. System model

We consider a multi-hop CR network formed by N nodes $1, 2, \cdots N$. These nodes are equipped with a half-duplex transceiver i.e., each node can either transmit or receive at any instant of time on a specified channel. These nodes are interested in point-to-point communication, i.e., we consider only the single hop communication. We assume that the spectrum is divided into M orthogonal channels $1, 2, \cdots M$ that are symmetric and assumed to be error-free. Further, we assume that a common control channel is available to exchange the control messages amongst the nodes. The nodes are time synchronized either through the common control channel or with the help of a GPS receiver. These assumptions of the system are kept similar to [1] to enable fair comparison between the heuristics proposed in this paper and the heuristics in phase 1 of [1]. Further we retained the phase 2 of [1] and is executed at the end of our allocation phase. We assume that all nodes have data to transmit to all its neighbors to ensure that our algorithm attacks the worst case scenario. Let us consider the MAC super frame which is shown in Fig. 2. The super frame is divided into sensing, information exchange, allocation and data transmission periods. In the sensing period, the channels that are free of primary is found. We assume perfect sensing by nodes for the sake of simplicity. The nodes send the list of channels available to a centralized repository or to their neighbors during the information exchange period. Then a spectrum allocation algorithm is executed and the schedule is distributed during the allocation period. The data transmission period is split into t equal length time-slots TS. The value of t is propagated through the network in phase 2. In a time-slot TS_k both the participating nodes transmit and receive data i.e., the time-slot is considered to consists of two sub slots where one node transmits and the second receives in the first sub slot, and in the second sub slot, the receiver in the first sub slot transmits its data to its counterpart. The sub slots may be of unequal lengths depending on the data that each node has to transmit to the other. This assumption reduces the complexity in the design, since each link has to be assigned with one time-slot. We list the parameters that are used in our algorithms in Table I.

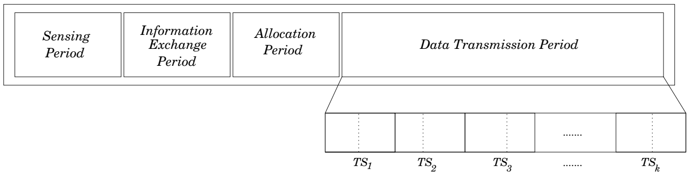

Fig. 2. MAC super frame

B. Problem description

We state the problem described in Section I here. Let the CRAN be represented by a graph $G = (V, E)$ where V is the set of vertices or nodes, and E is the set of links between vertices in V. A link m exists between two vertices u and v if both u and v are in communicating radii of each other and a

TABLE I
DEFINITIONS

G	Graph of CRAN, represented by adjacency matrix
N	Number of nodes in G
u or v	Represents a node in G
(u, v)	Represents the edge between u & v
M	Maximum number of channels in the CRAN

Notations	
$NodeID_u$	A unique ID of node u
δ_u	Degree of u
$Neighbors_u$	Set of neighbors of u
$Allocation$	A list containing slot & channel allocation for all edges
$Allocation_{(u,v)}.slot$	Slot allocated for (u, v)
$Allocation_{(u,v)}.channel$	Channel allocated for (u, v)
I	An Independent Set of G

common channel is available to them. The spectrum allocation problem is to find an optimal valid assignment

$$A_{(u,v),j,k} = \begin{cases} 1 & \text{if edge } (u,v) \text{ uses channel } j \text{ in time-slot } k \\ 0 & \text{otherwise} \end{cases}$$

The assignment is valid only if,

i) u and v are in communication radius of each other
ii) both u and v are not involved in any other communication in TS_k
iii) assignment of channel j does not cause interference to neighboring nodes.

It is necessary to find a minimum length MAC schedule for better channel reuse and in turn higher network throughput [8]. Finding the optimal MAC schedule can be proven to be $\mathcal{NP}$ complete.

IV. HEURISTICS FOR SPECTRUM ALLOCATION

We propose two heuristic algorithms based on: (i) Edge coloring and (ii) Clique determination. In the edge coloring algorithm, a central node learns the whole topology of the CRAN, performs the assignment and distributes the schedule to all the nodes. The central node can be a cluster-head [9]. In this network, the cluster-heads allocate, synchronize and maintain the network topology. The clique determination algorithm relaxes few assumptions considered above. There are no cluster-heads and each node is responsible for allocating a $(timeslot, channel)$ between its neighbors. A local common control channel is enough for nodes to learn about their neighbors and also employ a time synchronization scheme as in [10].

A. Edge Coloring Algorithm

The Edge coloring algorithm first calls Procedure 2 that edge colors G to assign time-slots and then assigns channels as described in Procedure 3. This algorithm produces a conflict-free schedule.

By performing an edge coloring to assign time-slots, it is ensured that no node is involved in more than one communication in the same time-slot. In this procedure, the first available channel for each edge is allocated such that the resulting assignment is valid as described in Section III-B.

In this algorithm we assume that each node knows schedules of its neighbors. A node that is not involved in communication during a slot will initiate communication with one of its neighbors who is also not involved in communication during that slot. The nodes can negotiate on a channel, if available, that does not cause interference to the assigned schedule. This 'free slot' utilization further increases the spectrum utilization.

The worst-case time complexity for the slot assignment procedure is $O(N^2)$ and for the channel assignment procedure is $O(MN)$.

Algorithm 1 Edge coloring algorithm

```
Input: Connectivity graph $G = (V, E)$ and the set of available channels $S_{(u,v)}$ between each pair of vertices $u, v$
Output: A valid allocation of $(timeslot, channel)$ for all edges
Run Procedure 2
Run Procedure 3
```

Procedure 2 Slot Assignment

```
repeat
    Find a node $u$ with maximum $\delta_u$ in $G$
    for $v$ in $Neighbors_u$ do
        a. $Allocation_{(u,v)}.slot \leftarrow$ first unused slot to edge $(u, v)$ such that the slot is unused in $u$ and $v$
        b. Remove the edge $(u, v)$ in $G$
    end for
    Mark $u$ as colored
until at least one node $u$ is uncolored
```

Procedure 3 Channel Assignment

```
$maxSlot \leftarrow$ maximum of $Allocation_{(u,v)}.slot \forall u, v$
repeat
    $k \leftarrow 1$
    for $(u, v)$ in $G$ such that $Allocation_{(u,v)}.slot = k$ do
        for $j$ in available channels for $(u, v)$ do
            if $Allocation_{(u,v)}.channel \leftarrow j$ is valid Assignment then
                a. $Allocation_{(u,v)}.channel \leftarrow j$
                b. Mark $(u, v)$ as assigned
            end if
        end for
        if $(u, v)$ is unassigned then
            a. $maxSlot \leftarrow maxSlot + 1$
            b. $Allocation_{(u,v)}.slot \leftarrow maxSlot$
        end if
    end for
until at least one edge $(u, v)$ is unassigned
```

B. Clique Determination Algorithm

In this algorithm, two hop topology information is exchanged between the nodes. In channel assignment algorithms, it is important to decide which node is responsible for assigning a $(timeslot, channel)$ to an edge to avoid multiple assignments yielding conflicts. To determine the responsibilities, each node executes the Procedure 5. At the end of the procedure, we have a set of nodes that form a maximal independent set (I) of the graph. Note that each node can independently determine which node in its 1-hop neighborhood is in the I with two hop topology information. The nodes in this set do not assign $(timeslot, channel)$ for any edge. That is the nodes in complement set, $\bar{I}$ assign $(timeslot, channel)$ to those edges for which the other end is in I or if other node has higher nodeID. Note that nodeID is used to rank the nodes in all the procedures. This is described in Procedure 6. Once the assignment is done, each node exchange the schedule in its two hop neighborhood. Conflicts may arise since each node executes the procedures locally. The conflicts are resolved as described in Procedure 7. If the conflict can be resolved by just switching to another channel, it is done so. In some cases, new slots have to be assigned. Since each node now has the knowledge of two hop neighbors' schedule, it picks out a conflict-free $(timeslot, channel)$ for the conflicted edge. An optimization procedure, optionally, can be executed by the nodes. In this procedure each node advertises its unused slots. If any other node can switch to one of those slots keeping the assignment valid, and the chosen slot is lower than the assigned slot, then that edge is assigned a new slot and channel. This reduces the schedule length, which is one of our goals. The 'free slot' procedure, as described in Section IV-A is also employed by the nodes to increase spectrum utilization. The worst-case time complexity for the I is $O(N)$, Assignment $O(MN)$, ConflictResolution is $O(N^2)$ at each node.

Algorithm 4 Clique based heuristic

```
Input:the set of one hop neighbors for each node $u$ and the set of available channels between each pair $u, v$ such that $v$ is a neighbor of $u$
Output:A valid allocation of $(timeslot, channel)$ for all edges of $u$
$I \leftarrow \emptyset$
Run Procedure 5
Run Procedure 6
Distribute $Allocation_u$ over one hop neighborhood
Run Procedure 7
```

Procedure 5 Form Independent Set

```
At each node $u$ do
$minDeg \leftarrow$ minimum $\delta_v \forall v$ in $Neighbors_u$
$minDegV \leftarrow v$ such that $\delta_v = minDeg$
if $\delta_u < minDeg$ then
  $I \leftarrow I$ U $\{u\}$
else if $\delta_u = minDeg$ and $u < minDegV$ then
  $I \leftarrow I$ U $\{u\}$
end if
```

Procedure 6 Slot and Channel Assignment

```
At each node $u \notin I$ do
for $v$ in $Neighbors_u$ do
  if $NodeID_u < NodeID_v$ or $v \in I$ then
    a. $Allocation_{(u,v)}.slot \leftarrow$ first unused slot in $u$
    b. $Allocation_{(u,v)}.channel \leftarrow$ an available channel for edge $(u, v)$
  end if
end for
```

Procedure 7 Conflict Resolution

```
At each node $u$ do
for $v$ in $Neighbors_u$ do
  if $Allocation_{(u,v)}.channel$ causes interference within 2-hop neighborhood then
    Find another channel $j$ for $(u, v)$ which results in valid assignment
    if no such $j$ exists then
      Allocate a new slot and channel for $(u, v)$ that results in valid assignment
    end if
  end if
end for
```

V. RESULTS AND DISCUSSIONS

In this section we evaluate the performance of the proposed heuristics with the one in [1]. We consider the hybrid ordering scheme in the heuristic of [1], since it is shown to have better performance than other ordering schemes. We refer to the heuristic in [1] as MAC Scheduling.

Simulations were done to evaluate the heuristics. In the experiments, areas of 100x100m, 200x200m, and 300x300m were chosen with nodes of 10, 40, and 90 randomly distributed in respective areas. The number of nodes were chosen to maintain the same density for different areas. A shadow-fading model was used to determine if nodes were in communication ranges of each other. Number of channels were chosen randomly between 1 and 25. The simulation environment for all experiments had 4 PUs that covered the entire area and occupied certain channels with exponential on-off distribution. This created dynamic topologies in the graph, with number of channels varying between neighbors every time. Twenty runs of each experiment were conducted and the averages are reported. Fig. 3 compares the schedule lengths produced by different heuristics. As evident from the figure, the Edge coloring based heuristic outperforms the other two heuristics. The Edge coloring heuristic avoids the conflicts since the whole topology information is known. The Clique based heuristic also performs as well as the Edge coloring, outperforming

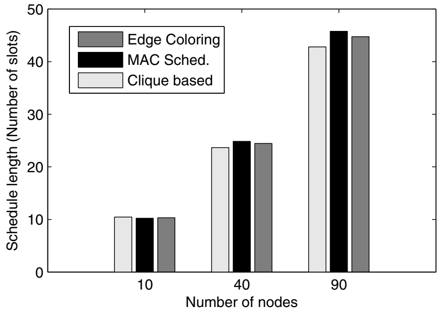

Fig. 3. Schedule lengths of different heuristics

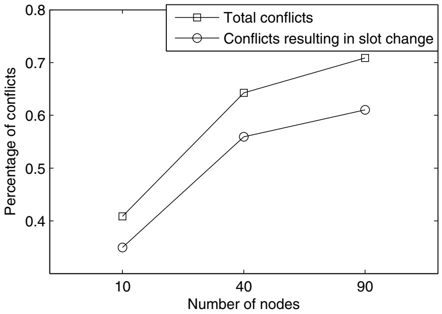

Fig. 4. Ratio of number of conflicts to edges vs number of nodes

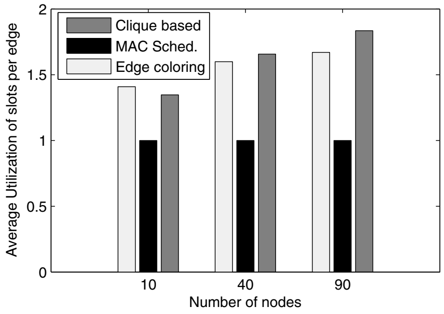

Fig. 5. Average utilization of edges from different algorithms

MAC Scheduling. Clique based heuristic considers the local the topology information to decide on $(timeslot, channel)$ pair for edges, and then resolves the conflicts. This iteration, not present in MAC Scheduling, tries to find a valid assignment without creating a new slot. Only if it fails, a new slot is created. Moreover the optimization procedure for Clique based heuristic that helps in reducing the schedule length. Fig. 4 shows the percentage of total conflicts that occur in the proposed Clique based heuristic versus different number of nodes in the network. The number of conflicts depends on the topology of the graph and the number of channels available between each pair of edges. Some of these conflicts can be resolved by switching to another available channel on one of the conflicted edges. The lower line in Fig. 4 shows the percentage of conflicts that required a slot change. This percentage is much smaller, which also explains the contribution of this iteration in producing of a smaller schedule length. Fig. 5 shows the improvement achieved in spectrum utilization due to the usage of 'free slots' for different algorithms. The MAC Scheduling heuristic is interested in only producing a schedule length, hence each edge receives only one slot. However in our proposed algorithms, we make use of free slots to increase spectrum utilization and hence some edges receive more number slots but it is ensured that each edge receives at least one slot. The average utilization depends on the graph under consideration. It is quite likely that the leaf edges receive more slots than the ones at the center. The Edge coloring heuristic knowing the complete topology at the central entity, it can schedule the usage of free slots. The Clique based heuristic has higher spectrum utilization since the schedule length is slightly higher than Edge coloring heuristic.

VI. CONCLUSIONS

In this paper, we considered the channel scheduling problem for CR network. We proposed two heuristics – Edge coloring and Clique determination based scheduling algorithms for CRANs. The proposed schemes also increased the spectrum utilization by using (slot, channel) pair that are being unused. Extensive experimentation results are presented. We also compared the results with an earlier work in [1] and the comparison shows significant performance improvement of the proposed schemes. The Edge based heuristic performs better than Clique based and [1]. The Clique based heuristic is better than MAC scheduling due to the additional iteration of conflict resolution procedure.

REFERENCES

[1] M. Thoppian, S. Venkatesan, R. Prakash, and R. Chandrasekaran, "MAC-Layer Scheduling in Cognitive Radio based Multi-Hop Wireless Networks," in *WOWMOM '06: Proceedings of the 2006 International Symposium on World of Wireless, Mobile and Multimedia Networks*. Washington, DC, USA: IEEE Computer Society, 2006, pp. 191–202.

[2] "FCC, Notice of proposed rule making and order, FCC ET Docket No 03-222," December 2003.

[3] "DARPA xG Program," http://www.darpa.mil/STO/Solicitations/WAND/pdf/XG_overview_for_WAND.pdf, 2007.

[4] H. Tamura, K. Watanabe, M. Sengoku, and S. Shinoda, "On a new edge coloring related to multihop wireless networks," in *Asia-Pacific Conference on Circuits and Systems, APCCAS*, vol. 2, 2002, pp. 357–360.

[5] I. Katzela and M. Naghshineh, "Channel assignment schemes for cellular mobile telecommunication systems: a comprehensive survey," *IEEE Personal Communications*, vol. 3, pp. 10–31, Jun 1996.

[6] S. Gandham, M. Dawande, and R. Prakash, "Link scheduling in sensor networks: distributed edge coloring revisited," in *Proceedings of IEEE INFOCOM*, vol. 4, March 2005, pp. 2492–2501.

[7] H. Zheng and C. Peng, "Collaboration and fairness in opportunistic spectrum access," in *IEEE International Conference on Communications, ICC*, vol. 5, 2005, pp. 3132–3136.

[8] S. Ramanathan and E. Lloyd, "Scheduling algorithms for multihop radio networks," *IEEE/ACM Transactions on Networking*, vol. 1, no. 2, pp. 166–177, 1993.

[9] T. Chen, H. Zhang, G. Maggio, and I. Chlamtac, "Topology management in cogmesh: A cluster-based cognitive radio mesh network," in *IEEE International Conference on Communications, ICC*, 2007, pp. 6516–6521.

[10] K. Römer, "Time synchronization in ad hoc networks," in *MobiHoc '01: Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing*, New York, NY, USA, 2001, pp. 173–182.