# Exploiting Multichannel Diversity for Cooperative Multicast in Cognitive Radio Mesh Networks

Hisham M. Almasaeid, *Member, IEEE* and Ahmed E. Kamal, *Fellow, IEEE*

***Abstract*—Cognitive radio networks (CRNs) have emerged as a promising, yet challenging, solution to enhance spectrum utilization, thanks to the technology of cognitive radios. A well-known property of CRNs is the potential heterogeneity in channel availability among secondary users. Therefore, multicast throughput in CRNs may suffer from significant degradation because of this property, since a link-level broadcast of a frame may only reach a small subset of destinations which are able to receive on the same channel. This may necessitate multiple sequential transmissions of the same frame by the source on different channels to guarantee delivery to all receivers in the destination set. In case of high data generation rate, delivery delay will be high due to the repeated transmissions by the source.**

**In this paper, we propose an assistance strategy to reduce the effect of the channel heterogeneity property on the multicast throughput in cognitive radio wireless mesh networks (CR-WMNs). This assistance strategy is composed of two main activities, first, allowing multicast receivers to assist the source in delivering the data, and second, allowing the transmission of coded packets so that multicast receivers belonging to different multicast groups can decode and extract their data concurrently. Results show that the proposed assistance paradigm reduces multicast time and increases throughput significantly.**

## I. Introduction

Empowered by the technology of software-defined radios [1], cognitive radio networks have emerged as a solution for the spectrum underutilization [2] problem. The new technology allows cognitive radio users, usually referred to as secondary users (SUs), to opportunistically utilize licensed spectrum bands when they are not being used by the licensed users, usually referred to as primary users (PUs). As the channel availability is both time and location dependent, SUs may observe heterogeneous sets of idle channels (this is referred to as the *heterogeneity property* of CRNs[1]) raising a number of challenges to the operation of CRNs.

In this paper, we are interested in the multicast problem in cognitive radio wireless mesh networks (CR-WMNs) [3]. Generally speaking, a wireless mesh network consists of a number of mesh routers (MRs) each of which manages a group of mesh clients (MCs) forming a cell. The mesh network is connected through a gateway to a backbone network, like the Internet. An MC may reach the backbone network through its parent MR first, and then through multiple hops of MRs until reaching the gateway. Multicasting is one of the important service modes, and is used in a number of networks to support many applications. In CR-WMNs, multicasting can be used when there are many groups of users, and members of each group are interested in receiving the same information. If these groups are in the same locality, the available bandwidth in unlicensed bands may not be sufficient, and opportunistic use of unused licensed spectrum may be the only means of supporting such applications. As an example, different groups of spectators in a sports event who would like to stream different replays of the game, stream broadcasts from other games being played at the same time, e.g., in the Olympics, or in the Soccer World Cup, or even receive information and statistics about the game(s), cannot be supported using the available ISM bands if the number of groups is large. Hence, CR-WMNs will enable the support of those users.

[1] The channel heterogeneity problem in traditional wireless networks, where all nodes see the same set of channels, refers to the unavailability of some channels due to contention. However, in cognitive radio networks the channel heterogeneity problem refers to different nodes seeing different sets of available channels. Nodes further contend for the available channels.

The heterogeneity property in CRNs may result in multicasting being implemented by transmitting the same frame multiple times, and over multiple channels, depending on the channel availability at different nodes. If the source is equipped with a single radio, these transmissions will have to be done sequentially, in order to deliver the same frame to all nodes in the receivers set. This may increase the time needed for a source to deliver the multicast frame to all receivers. In this work, we study the problem of minimizing the total multicast time in CR-WMNs by scheduling the multicast activity over both time and frequency and by also using the technique of network coding [4]. For this purpose, we propose a multicast mechanism that relies on three types of cooperation, namely; intra-group assistance, inter-group assistance, and the use of network coding. Three major operations facilitate these types of cooperation. The first operation is called *assistance*, in which some of the receiving members of a multicast group assist the multicast transmitter by forwarding the data on its behalf to other members of their group (or other groups). The second operation is called *overhearing*, in which some receiving members of a multicast group overhear the data destined to another group. This operation has two advantages; it first enables the inter-group assistance (forwarding) between different multicast groups, and also facilitates the delivery of multiple packets to different groups at the same time by using the third operation; the *codeword exchange operation*. In the *codeword exchange operation*, coded packets (or codewords) are used in the assistance operation so that members of different multicast groups can decode and extract their own data using packets they have previously overheard.

### A. The Multicast Scheduling Problem

Tremendous research has been conducted on multicast in multi-channel wireless networks to come up with efficient routing and/or channel allocation algorithms that maximize a number of different objectives. Energy-efficiency [5], spectrum efficiency [6], [7], throughput maximization [8], [9],

[10], [11], and delay minimization [8], [12] are examples of these objectives. Multicasting in CRNs is different from that in traditional multi-channel wireless networks. In traditional multi-channel wireless networks, the same set of frequency channels is available at all nodes. This assumption may not hold in CRNs due to the *heterogeneity property* mentioned earlier, as illustrated in the example in Fig. 1. The example shows three primary users $p_1$, $p_2$, and $p_3$, where each PU $p_i$ utilizes frequency channel $i$. The gray grid-line circle around each PU represents the protection range of that PU, within which no SU is allowed to concurrently utilize (transmit or receive) the frequency channel with the PU. This range may be determined based on different criteria. One criterion, for example, is to guarantee certain bit-error-rate performance for PUs. Also, six SUs exist in the network where one of them acts as a multicast source, while the others act as multicast receivers. Note how the geographical distribution of the nodes affects the channel availability at different SUs (represented by the set shown besides each SU). This difference forces the source SU to transmit the multicast data over the three frequency channels in order to cover all the multicast receivers.

The temporal part of the *heterogeneity property* is attributed to the channel usage distribution of PUs. For example, assume that at some point in time, $p_2$ is not using its frequency channel (i.e., channel 2). Then, all SUs will be able to use that channel, and consequently the source SU will be able to transmit the multicast data to all receivers over the same channel, i.e., channel 2. On the other hand, when the PU is back on the channel, SUs need to vacate it. This makes it a must for any scheduling algorithm to have a failure recovery plan. We refer the reader to [13] for more details this issue.

The time scale of channel availability is highly dependent on the wireless service, its coverage radius, spatial distribution of service users, and the temporal utilization of the service by those users. For example, TV channels which are not utilized in a particular region means that those channels are available all the time for SUs in that region. On the other hand, if the channels are utilized for a few hours during the day, then they are available for SUs during the rest of the day. Knowing the time span of both temporal and spatial channel availabilities requires conducting experiments in the area of interest to survey channel statistics [14], [15], [16] or estimating those time spans in real-time [17], [18].

### B. Contributions

We address the multicast scheduling problem in CR-WMNs in three phases. In the first phase, we study the scheduling of the multicast activity within a single cell, as defined in the Introduction. The contributions related to this phase are twofold. First, we proposed a scheduling strategy that exploits diversity in channel availability to enhance multicast throughput. Second, we proposed a centralized implementation of the proposed scheduling strategy within a single cell.

In the second phase, we study the issue of resolving potential conflict between the schedules of adjacent cells. We propose two solutions to guarantee collision-free schedule for the entire network; reactive collision resolution and proactive collision avoidance. In the last phase, we propose a recovery

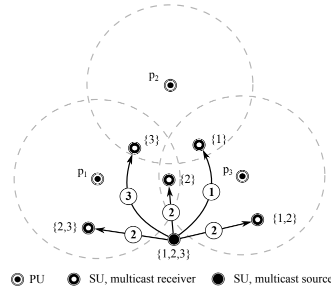



Fig. 1. An example that illustrates the heterogeneity property of CRNs.

algorithm to cope with the transmission failures due to PU activities. We finally evaluate the performance of the proposed algorithms and the effect of different network parameters on the achievable gain. Throughout this paper, we define gain as the percentage reduction in the multicast period. That is,

$$\text{gain} = \frac{multicast\ period_u \ - \ multicast\ period_a}{multicast\ period_u}\%$$

where $multicast\ period_a$ and $multicast\ period_u$ are the multicast periods with and without assistance, respectively.

It is worth pointing out that the multicast period here is the amount of time that an MR needs to deliver the multicast packets it has buffered to the intended receivers from among the MCs in its cell. In other words, it is the last hop delivery time, and not the end-to-end packet delivery time.

### C. Organization

We review the related work in Section II. The system model is presented in Section III. In Section IV, we formally define the *assisted multicast scheduling* problem and present some motivational examples. Then, in Section V, we elaborate on the problem complexity and propose ILP formulations for unassisted multicast scheduling and assisted multicast scheduling problems. A heuristic approach to solve the assisted multicast scheduling problem is proposed in Section VI. Resolving collisions between adjacent cells and recovering failed transmissions are studied in sections VII and VIII respectively. We evaluate the performance of the proposed algorithms in Section IX, and conclude in Section X.

## II. Related Work

In this section, we review related work. We first review the related work on multicast in CRNs. Then, we review the related work on network coding and its applications in CRNs.

### A. Multicast in Cognitive Radio Networks

In [19], subcarrier assignment and power allocation to multicast groups of SUs in OFDM-based CRNs, and under the constraint of tolerable interference to PUs, was studied. This work considers a single-cell wireless system in which a single base station (BS) serves both primary and secondary users. Subcarriers in the available spectrum that are not used for transmitting to PUs are exploited, by the BS, to serve SUs. As subcarriers may not necessarily be orthogonal, mutual interference between PUs and SUs on adjacent subcarriers was also considered. The results of this work can only be applied to scenarios in which the primary network manages the spectrum, through the BS, and determines what parts of the spectrum to sublease to SUs. Therefore, SUs cannot form their own network. Instead they have to be second-class users in an existing primary network.

The work in [20] presents a cooperative multicast scheduling scheme, including power control and channel allocation policies. In this scheme, a BS tunes its transmission power to avoid interference with PUs. SUs that are still reachable by the BS relay the multicast data through locally idle channels to unreachable SUs while tuning their transmission powers to avoid interference with PUs. Network coding is also used to reduce overhead and perform error control.

The joint problem of routing and channel assignment for multicast communication in multihop CRNs was addressed in a number of studies [21], [22], [23], [24]. The work in [21] tries to improve the scalability of the traditional ODMRP [25], in terms of number of multicast sources, in ad hoc wireless networks by using cognitive radios. An on-demand multicast routing algorithm for CR-WMNs was proposed in [22]. The algorithm finds the shortest path (in terms of number of hops) to the source of the multicast session, and optimally allocates frequency channels along that path to minimize the end-to-end delay using a dynamic programming approach. In [23], the problem of constructing a minimum-energy multicast tree in CRNs was studied. A routing and channel allocation algorithm based on a layered graph model was proposed in [24].

### B. Network Coding and its Applications in CRNs

Network coding (NC) [26], [4] has emerged as a very promising technique to enhance multicast throughput [27], [28] and provide protection and survivability [29], [30] in wireless networks. This technique allows nodes to combine packets instead of just forwarding them unchanged.

Very few studies have proposed applying NC to CRNs. Recently, a study was conducted in [31] to exploit network coding to increase the throughput of SUs in a CRN and decrease the amount of incurred interference and consumed energy. In a more recent study [32], the capacity of a cognitive radio relay network was investigated. Network coding is utilized to enable a cognitive system (pairs of SUs and sinks) to exchange information through a cooperative relay network that a primary system (pairs of PUs and sinks) is using to relay its own traffic. The effect of the interference of the cognitive system on the capacity of the primary system is studied. Moreover, the maximum achievable capacity for the cognitive system has also been studied. This work has shown that using network coding to make primary resources available for a cognitive system through cooperative relaying is viable and efficient.

Analog network coding (ANC) [33], a variant of network coding that operates at the signal level rather than the packet level, was used in [34] to enhance the throughput of Intercluster communication in CR-WMNs. Gateway nodes (those that are connecting clusters) exploit ANC to relay data between two adjacent clusters with minimum time, while at the same time control their transmission powers such that no harmful interference is caused to PUs. In this paper, we use the simplest form of network coding, that is the bitwise XORing of packets to achieve a further reduction in multicast time as will be explained later in the paper. Major differences between our work and that in literature are as follows:

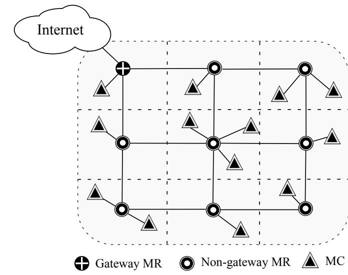


Fig. 2. Cognitive radio wireless mesh network model

- The solutions proposed in this paper are not restricted to any specific channel availability model.
- Our approach is simpler, easier to implement, and yet more optimal than other approaches which focus on making transmission decisions at the slot level rather than the frame level. Taking [20] as an example, a certain utility function is optimized given the history of previous decisions, which is not necessarily optimal. Also, the use of random network coding and coefficients from a large field makes the implementation more complex compared to the deterministic network coding we perform (using bit-wise XOR), and results in a greater overhead.
- Most of existing work, including [20], are more suited to temporal spectrum underutilization, while our approach is more suited to spatial spectrum underutilization.
- This work aims at minimizing multicast time (thus maximizing throughput) by exploiting the channel heterogeneity property of cognitive radio networks, which was not studied under traditional multichannel wireless networks.

## III. System and Service Models and Assumptions

We introduce here the system model and its assumptions.

### A. Network Model Assumptions:

- We consider a synchronized CR-WMN that operates in frames of time slots. It consists of a number of MRs connected in multiple hops to a gateway MR that provides access to the backbone network.
- Each MR, including the gateway MR, manages a set of mesh client (MCs) forming a cell.
- For cell $i$, let $\mathcal{A}_i = \{a_{0,i}, a_{1,i}, \cdots, a_{A_i,i}\}$ be the set of nodes (the MR and the MCs) in that cell, where $A_i$ is the total number of MCs in the cell (i.e., $A_i = |\mathcal{A}_i| - 1$ as $a_{0,i}$ is MR and the rest are MCs). An MC can access the backbone network only through its parent MR.
- The CR-WMN coexists with a primary network that utilizes a set of orthogonal channels $\mathcal{L}$.
- If a PU is active, and a transmission by one or more SUs may interfere with a transmission by this PU, then the PU's channel is considered unavailable to those SUs.
- If an SU can transmit and receive on a channel that is licensed to a certain PU without causing interference to it, then the channel is considered available to the SU.
- $\mathcal{L}_i$ denotes the set of available data channels at node $i$, where $\mathcal{L}_i \subseteq \mathcal{L}$.
- We assume that each MC shares at least one data channel with the parent MR of the cell that the MC belongs to.
- Each node has a single radio for data transmission.

- Secondary users (MRs and MCs) obtain the set of available channels (those which can be used without harming PUs) through spectrum sensing using any of the techniques proposed in literature, e.g., [35], [36], [37].
- We assume that the channel availability at SUs is quasi-static, i.e., does not change in a short period of time[2].
- We assume the presence of a Common Control Channel (CCC) on which nodes can exchange control information. This information includes channels availabilities at MCs, which are sent uplink to the MR, and the transmission schedule, including channel assignment, which is sent downlink from the MR to the MCs.

It is important to note that our assumption of an available channel $k$ at node $i$ means that node $i$ can transmit and receive on $k$ without interference to or from the licensee PU of $k$.

### *B. Multicast Service Assumptions:*

- We concentrate on the multicasting service mode, where data needs to be delivered to a predefined set of nodes.
- A multicast session originates at the gateway and it can span multiple cells. However, we are interested in multicast scheduling within individual cells, which will be treated independently, and also in avoiding collisions in schedules of adjacent cells.
- In a cell, $i$, the MR may need to serve $M$ multicast sessions, where each multicast session, $k$, is identified by a set of receiving MCs, $\{\mathcal{G}_{k,i}\}$. All MCs in the same session should receive the same set of data units.

We therefore treat the multicast process as a two-stage process. The first stage is to deliver the multicast data from the gateway to the MRs which have some of their MCs subscribing to the intended multicast session(s). The second stage is for each MR to deliver the received multicast data to the subscribing MCs within its cell. As stated earlier, we are concerned with the second stage, and our objective is to devise collision-free multicast schedules while minimizing the total multicast time within a single cell. We are also interested in having collision free scheduling between adjacent cells.

## IV. Motivation and Problem Definition

Before we formally define the assisted multicast problem, we would like to present an example that illustrates the motivation behind this work, and then give some definitions.

### *A. Motivational Example*

Consider the network (a single cell) in Fig. 3. The figure shows two multicast groups: the white MC nodes form the group $\mathcal{G}_1 = \{n_1, n_2, n_3, n_4, n_6\}$ that should receive packet $a$, and the gray MC nodes form the group $\mathcal{G}_2 = \{n_5, n_7, n_8\}$ that should receive packet $b$. The set besides each MC represents the channels available to that MC. Node $n_0$ represents the MR of the cell, and it has all the five channels available (following the CM channel availability model). Table I, summarizes the basic idea of assisted multicast for the network in Fig. 3. The first two rows show an optimal multicast schedule without any form of assistance, the first of the two shows the transmissions as $(transmitter, packet, channel)$ tuples, and the second shows the receptions as $(receivers, packet, channel)$ tuples. Similar pairs of rows are presented for three levels of assistance, each of which corresponds to exploiting an additional assistance operation. Columns in Table I correspond to time slots. As the table explains, under no form of assistance, the best the MR can do is 6 time slots. By adding intra-group assistance, i.e., allowing members of the same group to forward packets to each other, the total multicast time was reduced to 5 slots. By extending the assistance to inter-group [3] the total time was further reduced to 4 time slots. Finally, by allowing nodes to exchange coded (bitwise *XOR*ed) packets, the total time was reduced to 3 slots.

In the schedule in the last pair of rows (which uses the three levels of assistance), note that MCs $n_1$ and $n_6$ (interested in packet $a$) have received packet $b$ in slot $T_1$, and MC $n_5$ (interested in packet $b$) has received packet $a$ in slot $T_2$. Therefore, all the three MCs will be able to decode the $a \oplus b$ packet they received in slot $T_3$ and extract their own data. This schedule is presented in Fig. 3. It is worth pointing out that scheduling the overhearing opportunities can highly affect the achievable gain.transmissions during a single time slot.

### *B. Definitions*

We present, in this subsection, some necessary definitions.

***Definition** 4.1: Codeword*: is a group of packets (could be a single packet) coded (bitwise *XOR*ed) into one packet.

***Definition** 4.2: Assistance operation*: is the process of having one MC forward to another MC in slot $t$ a codeword that the latter can use, possibly with the codewords it has overheard in $[0, t-1]$, to extract the data destined to it. If the two MCs belong to the same group, then this operation is called *Intra-group assistance*. Otherwise, it is called *Inter-group assistance*.

***Definition** 4.3: Codeword exchange*: is the process of allowing the exchange of codewords in the *assistance operation*.

***Definition** 4.4: Multicast period*: is the number of time slots needed by the MR to deliver the data packet destined to a multicast group to all the members of that group.

***Definition** 4.5: Multicast schedule*: is a schedule of the multicast activity over time and frequency. The schedule should determine for each member of a multicast group (including the MR) what to transmit/receive (codeword), on what frequency (channel), and at what time (slot). A multicast schedule is feasible *iff* the following conditions are satisfied.

1) *Interference constraint*: at any slot $t$, there can be at most one transmission per channel (We assume that all the nodes of a cell (MCs and the MR) are within the interference range of each other).
2) *Radio constraint*: at any slot $t$, there can be at most one transmission per node (single radio/node).

[2]We consider environments in which the primary users, when they become idle they do so for a long time (i.e., channels availability dynamics are very slow). If channel status changes frequently with short inter-change time, the availability of channel information will impose strong requirements on spectrum sensing. This will limit the applicability of the proposed algorithms

[3]Due to its advantages of enhancing throughput,and reducing delay, inter-group assistance, in which receivers from one group assist members of other groups, can be motivated by using a strategy like Tit-for-Tat, which is widely used in Peer-to-Peer network protocols.

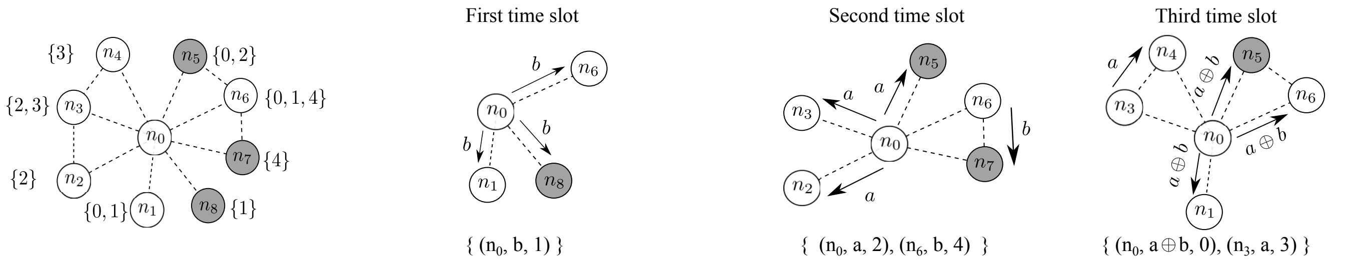



Fig. 3. An example that shows the benefit of using assisted multicast in reducing the total multicast period.

TABLE I
ENHANCING THROUGHPUT BY INTRODUCING DIFFERENT ASSISTANCE MECHANISMS.

| Scenario | Tx/Rx | $T_1$ | $T_2$ | $T_3$ | $T_4$ | $T_5$ | $T_6$ |
|---|---|---|---|---|---|---|---|
| Unassisted multicast | *Tx* | $(n_0, a, 0)$ | $(n_0, a, 2)$ | $(n_0, a, 3)$ | $(n_0, b, 0)$ | $(n_0, b, 4)$ | $(n_0, b, 1)$ |
| | *Rx* | $(\{n_1, n_6\}, a, 0)$ | $(\{n_2, n_3\}, a, 2)$ | $(n_4, a, 3)$ | $(n_5, b, 0)$ | $(n_7, b, 4)$ | $(n_8, b, 1)$ |
| Intra-group assis. | *Tx* | $(n_0, a, 0)$ | $(n_0, a, 2)$ | $(n_3, a, 3)$, $(n_0, b, 0)$ | $(n_0, b, 4)$ | $(n_0, b, 1)$ | - |
| | *Rx* | $(\{n_1, n_6\}, a, 0)$ | $(\{n_2, n_3\}, a, 2)$ | $(n_4, a, 3)$, $(n_5, b, 0)$ | $(n_7, b, 4)$ | $(n_8, b, 1)$ | - |
| Inter-group assis. | *Tx* | $(n_0, b, 1)$ | $(n_0, a, 2)$, $(n_6, b, 4)$ | $(n_3, a, 3)$, $(n_0, b, 0)$ | $(n_0, a, 0)$ | - | - |
| | *Rx* | $(\{n_8, n_6\}, b, 1)$ | $(\{n_2, n_3\}, a, 2)$, $(n_7, b, 4)$ | $(n_4, a, 3)$, $(n_5, b, 0)$ | $(\{n_1, n_6\}, a, 0)$ | - | - |
| Codeword exchange | *Tx* | $(n_0, b, 1)$ | $(n_0, a, 2)$, $(n_6, b, 4)$ | $(n_3, a, 3)$, $(n_0, a \oplus b, 0)$ | - | - | - |
| | *Rx* | $(\{n_1, n_6, n_8\}, b, 1)$ | $(\{n_2, n_3, n_5\}, a, 2)$, $(n_7, b, 4)$ | $(n_4, a, 3)$, $(\{n_1, n_5, n_6\}, a \oplus b, 0)$ | - | - | - |

3) *Precedence constraint*: For an MC to transmit codeword $v$ at time $t$, it must receive a set of codewords in $[1, t-1]$ sufficient to construct $v$.

Then, the assisted multicast scheduling (AMS) problem in CR-WMNs is defined as follows:

***Definition** 4.6: AMS problem in CR-WMNs*: Given $M$ multicast groups $\{\mathcal{G}_{1,i}, \cdots, \mathcal{G}_{M,i}\}$ managed by MR $a_{0,i}$ in cell $i$, find a feasible *multicast schedule* within the cell, with both *assistance* and *codeword exchange* operations enabled, that results in the minimum *multicast period*.
Table II summarizes all the notations of this paper.

## V. Problem Complexity and Formulation

In this section, we study the complexity of the AMS problem and propose two integer linear programs (ILP's) for the cases of unassisted multicast and single multicast group with intra-group assistance, respectively.

### *A. Single multicast group complexity*

We first consider the case of a single multicast group in a single cell of the CR-WMN. In such a case, the only possible form of assistance is the intra-group assistance between the members of the multicast group. To understand the complexity of the *"AMS for a single group"* problem, let us study that of the normal, unassisted, multicast scheduling problem as the latter is a special case of the former.

***Definition** 5.1: Unassisted Multicast scheduling for a single group (UMS-Single)*: for a cell $i$, given a single multicast group $\mathcal{G}_{j,i}$ managed by MR $a_{0,i}$ and the set of available channels for each node $u \in \mathcal{G}_{j,i} \cup \{a_{0,i}\}$. Find a *multicast schedule* that results in the minimum *multicast period* given that $a_{0,i}$ is the only transmitter (i.e., no assistance).

***Theorem** 5.1:* The *UMS-Single* problem is NP-hard.

*Proof:* See Appendix A. ∎

As the UMS-Single is a special case of the AMS-Single, the latter is also NP-hard. In other words, any instance of the UMS-Single problem can be mapped into an instance of the AMS-Single problem with all edges between MCs removed (to prevent assistance between MCs). Next, we present two ILP formulations for the two problems. These ILPs will be used in Section IX to evaluate the gain of using the assistance operation. Before giving the ILPs, we present some notations:

- $T_{j,i}$ is the maximum number of time slots needed to deliver the multicast packet of group $\mathcal{G}_{j,i}$ to all of its members. $T_{j,i} = \min\{|\mathcal{L}_{a_{0,i}}|, |\mathcal{G}_{j,i}|\}$.
- $\nu^t$ is a binary variable that is set to 1 if a transmission exists in slot $t$ on any of the channels in $\mathcal{L}$.
- $y^t_{u,k}$ is a binary variable that, if set to 1, means that $u$ transmits on channel $k$ at slot $t$.

### *B. ILP for the UMS-Single problem*

The UMS-Single problem for a cell $i$ is formulated as follows, where $\hat{u}$ is the MR of $i$ (i.e., $a_{0,i}$).

**ILP-UMS**: Minimize $\sum_{t=1}^{T_{j,i}} \nu^t$, subject to:

$$\sum_{k \in \mathcal{L}_{\hat{u}}} y^t_{\hat{u},k} \leq \nu^t, \qquad 1 \leq t \leq T_{j,i} \quad (1)$$

$$\sum_{k \in \mathcal{L}_u \cap \mathcal{L}_{\hat{u}}} \sum_{\tau=1}^{T_{j,i}} y^{\tau}_{\hat{u},k} \geq 1, \qquad u \in \mathcal{G}_{j,i} \quad (2)$$

The objective is to minimize the total number of used time slots. Since the system operates in cycles (or time frames), where in each cycle the same schedule obtained from the optimization formulation above is used, then minimizing the number of time slots within a cycle is equivalent to maximizing the system throughput. Constraint (1) guarantees at most one transmission per time slot. Constraint (2) guarantees that

TABLE II
SUMMARY OF NOTATIONS

| | |
|---|---|
| $\mathcal{A}_i$ | $\mathcal{A}_i = \{a_{0,i}, \cdots, a_{A_i,i}\}$ is the set of nodes in cell $i$. $a_{0,i}$ is the MR managing the cell, and $a_{1,i}, \cdots, a_{A_i,i}$ are the MCs in cell $i$. $A_i{=}\lvert\mathcal{A}_i\rvert{-}1$ is the number of MCs in $i$. |
| $c(u)$ | the cell to which node $u$ belongs |
| $\mathcal{S}$ | the set of multicast sessions in the network, $\lvert\mathcal{S}\rvert = M$. |
| $p(j)$ | the current packet to be delivered of multicast session $j$ |
| $\mathcal{G}_{j,i}$ | the set of multicast receivers of session $j$ in cell $i$ |
| $\mathcal{G}_{j,i}$ | the set of multicast receivers of session $j$ in cell $i$ that have received packet $p(j)$ |
| $\mathcal{G}_i = \cup_{j=1}^{\lvert\mathcal{S}\rvert} \mathcal{G}_{j,i}$ | the set of all multicast receivers of all sessions in cell $i$ |
| $\mathcal{N}_{j,i}(u)$ | the set of neighbors of node $u$ in $\mathcal{G}_{j,i} \cup \{a_{0,i}\} \backslash \{u\}$ |
| $\mathcal{N}_i(u)$ | the set of neighbors of node $u$ in $\mathcal{G}_i \backslash \{u\}$ |
| $\mathcal{L}$ | the set of licensed channels opportunistically utilized by the CR-WMN, such that $\lvert\mathcal{L}\rvert = K$ |
| $\mathcal{L}_j \subseteq \mathcal{L}$ | the set of channels available to node $j$ |
| $p_{ON}$ | the probability of a PU being active, i.e., using its channel |
| $\zeta$ | the interference range |
| $V_u$<br>$\mathcal{V}_u$ | the set of codewords overheard by MC $u$<br>the set of all combinations of the codewords in $V_u$ |
| $\mathcal{X}_i[t]$ | the set of multicast transmissions scheduled in cell $i$ in slot $t$. Each transmission $x{\in}\mathcal{X}_i[t]$ is represented by the tuple $(z, v, k, \mathcal{R})$, where $z$ is the transmitter, $v$ is the codeword, $k$ is the channel, and $\mathcal{R}$ is the set of receivers. |
| $\mathcal{X}_i$ | the multicast schedule of cell $i$ such that $\mathcal{X}_i = \{\mathcal{X}_i[1], \cdots, \mathcal{X}_i[t], \cdots, \mathcal{X}_i[T_f]\}$, where $T_f$ is the frame length. |

every MC will receive the data by forcing the MR to transmit on at least one of the channels available to that MC.

### C. ILP for the AMS-Single problem

The ILP presented for the UMS-Single problem can be modified to formulate the AMS problem with intra-group assistance for a single multicast group. We just need to allow MCs to forward the data they receive to their neighbors. The ILP formulation of the AMS-Single problem is as follows:

**ILP-AMS**: Minimize $\sum_{t=1}^{T_{j,i}} \nu^t$, subject to:

$$\sum_{k \in \mathcal{L}_u} y_{u,k}^t \le \nu^t, \qquad u \in \mathcal{G}_{j,i} \cup \{a_{0,i}\}, 1 \le t \le T_{j,i} \quad (3)$$

$$\sum_{k \in \mathcal{L}_u} y_{u,k}^1 = 0, \qquad u \in \mathcal{G}_{j,i} \quad (4)$$

$$\sum_{k \in \mathcal{L}_u} y_{u,k}^t \le \sum_{w \in \mathcal{N}_{j,i}(u)} \sum_{k \in \mathcal{L}_u \cap \mathcal{L}_w} \sum_{\tau=1}^{t-1} y_{w,k}^\tau, \quad u \in \mathcal{G}_{j,i}, 2 \le t \le T_{j,i} \quad (5)$$

$$\sum_{w \in \mathcal{N}_{j,i}(u)} \sum_{k \in \mathcal{L}_u \cap \mathcal{L}_w} \sum_{\tau=1}^{T_{j,i}} y_{w,k}^\tau \ge 1, \qquad u \in \mathcal{G}_{j,i} \quad (6)$$

$$\sum_{u \in \mathcal{G}_{j,i} \cup \{a_{0,i}\} : k \in \mathcal{L}_u} y_{u,k}^t \le 1, \qquad k \in \mathcal{L}, 1 \le t \le T_{j,i} \quad (7)$$

Constraint (3) guarantees that at most one transmission per node (MR or MC) exists in a time slot. Using constraint (4), we forbid MCs from transmitting in the first time slot as they have not received the multicast packet yet. Constraint (5) guarantees that no MC transmits on any channel at slot $t$ before it receives the packet from at least one neighbor, on a channel common between the two, in $[1, t{-}1]$. We guarantee the delivery of the multicast packet to each MC by constraint (6). Constraints (7) and (3) guarantee one transmission per channel and one transmission per node in each time slot respectively.

### D. The complexity of the AMS problem with multiple groups

Apparently, the AMS problem with multiple multicast groups is at least as hard as the AMS with a single group, which is NP-hard as proved in the previous subsection. In fact, the ILP formulation of the AMS problem with multiple groups is more complicated because of the *codeword exchange operation*. Specifically, an MC cannot transmit a codeword $v$ at time $t$ unless it receives a set of codewords sufficient to construct $v$. To embed this fact into the ILP, we need to take into consideration all combinations of native multicast packets which will increase the number of variables and constraints exponentially. Moreover, the constraint which ensures that each MC receives its multicast packet is also more complicated. Instead of a unique packet that satisfies the constraint in the case of a single group, a group of decodable codewords can satisfy the delivery constraint in the case of multiple groups with the *codeword exchange operation*. This requires us to take into account all the combinations of decodable codewords from which an MC can extract its packet. This increases the number of constraints exponentially. Therefore, no ILP is proposed for the AMS problem with multiple groups.

## VI. HEURISTIC SOLUTION FOR THE AMS PROBLEM

In this section, we propose a heuristic algorithm to solve the AMS problem with multiple multicast groups. The algorithm is greedy-based in the sense that it deals with each slot independently and tries to make the optimal decision at this slot. However, finding this optimal decision in each time slot is not an easy task. In fact, it can be shown that for the case of a single multicast group, scheduling the transmissions of the MR and covered MCs at a time slot $t$ (those which have received the multicast packet in $[1, t-1]$) in a slot $t$ such that the packet is delivered to the maximum number of uncovered MCs is NP-hard (assuming of course that covered MCs may *assist* uncovered ones). Therefore, we divide the scheduling task in *a single time slot* $t$ into three phases.

- **Phase-1**: Scheduling the MR transmission (what codeword to transmit, and on which channel).
- **Phase-2**: Scheduling the assistance operation for each assistance candidate (what codeword to transmit and on which channel). An *assistance candidate* is an MC that was not scheduled to receive data in the first phase, and has received at least one codeword in $[1, t-1]$.
- **Phase-3**: Scheduling overhearing opportunities for overhearing candidates. An *overhearing candidate* is an MC that was not scheduled as a transmitter (assistant MC) or a receiver in the first two phases. Such an MC has the choice to overhear any of the scheduled codeword transmissions it can. It shall overhear the codeword transmission that has the highest potential of being beneficial to the MC itself or any of its neighbors later.

Note that all these operations are scheduled over frequency channels only and not over time. Before presenting the details of each of the three phases, we provide some terminology.

Let $V_u$ be the set of overheard codewords by MC $u$ up until the current time slot. Also, let $p(j)$ be the current packet to be delivered of multicast session $j$, and let an MC $u \in \mathcal{G}_{j,c(u)}$ where $c(u)$ is the cell to which $u$ belongs. Assume that $V_u$

does not produce $p(j)$, i.e., no combination of the codewords in $V_u$ can produce $p(j)$. Then, the set of useful codewords to $u$ (those that $u$ can use along with $V_u$ to decode and extract $p(j)$ from) can be determined as follows. Define $\mathcal{V}_u$ as the set of all combinations, bitwise XORs, of the codewords in $V_u$, i.e., $|\mathcal{V}_u| = 2^{|V_u|} - 1$. Then, $p(j) \oplus \mathcal{V}_u^l$ is a useful codeword for MC $u$, where $\mathcal{V}_u^l$ denotes the $l^{th}$ combination (codeword) of $\mathcal{V}_u$, for all $1 \leq l \leq |\mathcal{V}_u|$.

Let us consider **Phase-1**, and let $\overline{\mathcal{G}}_{j,i} \subseteq \mathcal{G}_{j,i}$ be the set of MCs in cell $i$ that belong to session $j$ and have received $p(j)$. Then, the best $(codeword, channel)$ schedule for the MR $\hat{u}$ that belongs to cell $i$ in a given time slot is found as follows:

$$(v^*, k^*) = \underset{(v\in\mathcal{V},k\in\mathcal{L}_{\hat{u}})}{\operatorname{argmax}} \sum_{j\in\mathcal{S}} \sum_{u\in\mathcal{G}_{j,i}\backslash\overline{\mathcal{G}}_{j,i}} \begin{cases} 1 & k\in\mathcal{L}_u,\ v{=}p(j) \\ & \text{or } v\oplus p(j){\in}\mathcal{V}_u \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

where $\mathcal{S}$ is the set of multicast sessions, and $\mathcal{V}$ is defined as,

$$\mathcal{V} = \left\{ p(j) \cup \bigcup_{l=1}^{|\mathcal{V}_u|} p(j) \oplus \mathcal{V}_u^l : \forall j \in \mathcal{S},\ u \in \mathcal{G}_{j,i}\backslash\overline{\mathcal{G}}_{j,i} \right\} \tag{9}$$

Equation (8) finds, for the MR, the $(codeword, channel)$ pair that serves the maximum number of unserved MCs at a particular time slot[4]. The same approach is used for the second phase, namely, scheduling the assistance operation. For an assistance candidate MC $u$, where $c(u) = i$, the optimal $(codeword, channel)$ in a time slot $t$ is found as follows:

$$(v_u^*, k_u^*) = \underset{(v\in\mathcal{V}_u,k\in\mathcal{L}_u/\mathcal{K}[t])}{\operatorname{argmax}} \sum_{j\in\mathcal{S}} \sum_{w\in\mathcal{G}_{j,i}\backslash\overline{\mathcal{G}}_{j,i}} \begin{cases} 1 & \text{If } k\in\mathcal{L}_w,\ v{=}p(i) \\ & \text{or } v{\oplus}p(j){\in}\mathcal{V}_w, \\ & w{\in}\mathcal{N}_i(u) \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

where $\mathcal{K}[t]$ is the set of busy channels in time slot $t$.

The last phase is to schedule the overhearing operation for MCs that are not receiving data or participating in the assistance operation. The basic idea is for an MC to overhear the codeword that is *useful* to the maximum number of its neighbors. Let $\mathcal{X}_i[t]$ be the set of all transmissions in cell $i$ at time slot $t$, represented as $(transmitter, codeword, channel, receivers)$ tuples. Let $z_l$, $v_l$, $k_l$, and $\mathcal{R}_l$ denote the transmitter, codeword, channel, and set of receivers of multicast transmission $x_l$. Then, for an *overhearing candidate* MC $u$, where $c(u){=}i$, the best transmission $x^*{=}(z_u^*, v_u^*, k_u^*, \mathcal{R}_u^*)$ to overhear is given as:

$$(z_u^*, v_u^*, k_u^*, \mathcal{R}_u^*) = \underset{(z,v,k,\mathcal{R})\in\mathcal{X}_i[t]}{\operatorname{argmax}} \sum_{j\in\mathcal{S}} \sum_{w\in\mathcal{G}_{j,i}\backslash\overline{\mathcal{G}}_{j,i}} \begin{cases} 1 & \text{If } v{\in} \bigcup_{l=1}^{|\mathcal{V}_w|} \mathcal{V}_w^l \oplus p(j), \\ & w \in \mathcal{N}_i(u) \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

The AMS heuristic approach, denoted HAMS, is outlined in Algorithm 1. The first phase, i.e., scheduling the MR transmission, is expressed by lines $[7-15]$. The phase of scheduling the assistance operation is expressed by lines $[18-32]$. The phase of scheduling overhearing opportunities is expressed in the loop starting at line 33. Finally, unnecessary scheduled overhearings (those which were not used to decode any useful packet) are removed at line 37.

[4]Note that $u \in \mathcal{G}_{j,i}\backslash\overline{\mathcal{G}}_{j,i}$ implies, by definition, that $p(j) \notin \mathcal{V}_u$ in (8).

**Algorithm 1**: HAMS: Heuristic solution for the AMS problem for cell $i$.

**input** : Multicast groups $\{\mathcal{G}_{1,i}, \cdots, \mathcal{G}_{M,i}\}$, $\mathcal{L}_i\ \forall i \in \mathcal{G}_i \cup \{a_{0,i}\}$.

1 $V_u \leftarrow \emptyset\ \forall u \in \mathcal{G}_i,\ \overline{\mathcal{G}}_{j,i} \leftarrow \emptyset\ \forall j \in \mathcal{S},\ t = 0;$
2 **while** $\exists j \in \mathcal{S} : |\overline{\mathcal{G}}_{j,i}| < |\mathcal{G}_{j,i}|$ **do**
3 $t \leftarrow t + 1;$
4 $\mathcal{B}[t] \leftarrow \emptyset$; *//Busy MCs in slot $t$*
5 $\mathcal{X}_i[t] \leftarrow \emptyset$; *//Transmissions in slot $t$*
6 $\mathcal{K}[t] \leftarrow \emptyset$; *//Busy channels in slot $t$*
7 Find the optimal $(codeword, channel)$ for the MR using eq. (8), let that be $(v^*, k^*)$ ;
8 $\mathcal{R} \leftarrow \emptyset;$
9 **forall** $(j, u) : j \in \mathcal{S},\ u \in \mathcal{G}_{j,i}\backslash\overline{\mathcal{G}}_{j,i}, k^*{\in}\mathcal{L}_u,\ v^* \oplus p(j) \in \mathcal{V}_u$ **do**
10 $\mathcal{R} \leftarrow \mathcal{R} \cup \{u\};$
11 $V_u \leftarrow V_u \cup \{v^*\};$
12 $\overline{\mathcal{G}}_{j,i} \leftarrow \overline{\mathcal{G}}_{j,i} \cup \{u\};$
13 $\mathcal{B}[t] \leftarrow \mathcal{B}[t] \cup \{a_{0,i}\} \cup \mathcal{R};$
14 $\mathcal{K}[t] \leftarrow \mathcal{K}[t] \cup \{k^*\};$
15 $\mathcal{X}_i[t] \leftarrow \mathcal{X}_i[t] \cup \{(a_{0,i}, v^*, k^*, \mathcal{R})\};$
16 $\mathcal{R} \leftarrow \emptyset;$
17 *//Schedule the assistance operation*
18 **while** $|\mathcal{G}_i\backslash\mathcal{B}[t]| > 2$ **do**
19 **forall** $u \in \mathcal{G}_i\backslash\mathcal{B}[t]$ **do**
20 Find the optimal $(codeword, channel)$ for MC $u$ using eq. (10) and let that be $(v_u^*, k_u^*)$, and let the value of the maximum be $\alpha_u^*$;
21 $\hat{u} = \underset{u\in\mathcal{G}_i\backslash\mathcal{B}[t]}{\operatorname{argmax}}\ \alpha_u^*;$
22 **if** $\alpha_{\hat{u}}^* = 0$ **then**
23 break;
24 **else**
25 $\mathcal{R} \leftarrow \{u : \exists j \in \mathcal{S} \text{ where } u \in \left(\mathcal{G}_{j,i}\backslash\overline{\mathcal{G}}_{j,i}\right) \cap \left(\mathcal{N}_i(\hat{u})\backslash\mathcal{B}[t]\right), v_{\hat{u}}^* \oplus p(j) \in \mathcal{V}_u, k_{\hat{u}}^* \in \mathcal{L}_u\};$
26 $\mathcal{X}_i[t] \leftarrow \mathcal{X}_i[t] \cup \{(\hat{u}, v_{\hat{u}}^*, k_{\hat{u}}^*, \mathcal{R})\};$
27 $\mathcal{B}[t] \leftarrow \mathcal{B}[t] \cup \{\hat{u}\} \cup \mathcal{R};$
28 $\mathcal{K}[t] \leftarrow \mathcal{K}[t] \cup \{k_{\hat{u}}^*\};$
29 **forall** $(j, w) : j \in \mathcal{S},\ w \in \mathcal{N}_i(\hat{u}) \cap \mathcal{G}_{j,i}\backslash\overline{\mathcal{G}}_{j,i},\ k_{\hat{u}}^* \in \mathcal{L}_w,\ v_{\hat{u}}^* \oplus p(j) \in \mathcal{V}_w$ **do**
30 $\overline{\mathcal{G}}_{j,i} \leftarrow \overline{\mathcal{G}}_{j,i} \cup \{w\};$
31 $\mathcal{B}[t] \leftarrow \mathcal{B}[t] \cup \{w\};$
32 $V_w \leftarrow V_w \cup \{v_u^*\};$
33 **forall** $u \in \mathcal{G}_i\backslash\mathcal{B}[t]$ **do**
34 Find the optimal transmission $x^* = (z^*, v^*, k^*, \mathcal{R}^*) \in \mathcal{X}_i$ for MC $u$ to overhear using eq. (11);
35 $V_u \leftarrow V_u \cup \{v_u^*\};$
36 Add $u$ to the receivers, i.e. $\mathcal{R}^*$, of the multicast transmission.
37 Remove unused overhearings for all MCs;

## VII. Collision-Free Scheduling Across Cells

So far, we have been concerned with scheduling the multicast activity within a single cell. The potential conflict between adjacent cells, due to collisions, was not taken into consideration. In this section, we investigate possible solutions to prevent conflicts between the schedules of adjacent cells and limit the effect of such conflict on the gain achieved by the proposed assistance mechanism. We will investigate two approaches to avoid/resolve conflicts; a *proactive* approach that guarantees conflict-free schedules, and a *reactive* approach that allows conflicts, then resolves them after they are detected.

### A. Proactive approach

Under this approach, whenever an MR calculates the schedule of the cell it manages, it informs all the adjacent cells about the channels it uses in each time slot. Therefore, any MR that needs to calculate/update the multicast schedule of the cell it manages must refrain from using any channel during a particular slot $t$ that an adjacent cell is using in that slot. In other words, when an MR uses the HAMS algorithm to calculate the schedule, it shall add to the set of busy channels in slot $t$, i.e., $\mathcal{K}[t]$, all the channels that are used by adjacent cells in slot $t$. When a particular channel is no longer used in a specific time slot in a cell, the MR managing that cell must inform adjacent cells about this change.

This approach guarantees collision free schedules. Therefore, no post-scheduling phase is needed. Moreover, it is simple to implement. However, this approach may limit the potential gain of the proposed assistance mechanism because some cells may not be able to utilize the full set of channels available to its nodes (to avoid collisions with adjacent cells). Furthermore, maintaining up-to-date channel usage information across adjacent cells incurs communication overhead.

This approach is outlined in Algorithm 2. When an MR is ready to activate the schedule of the cell it manages, it sends a scheduling request to all adjacent cells (line 2). This request will help us resolve concurrent activation of schedules which may result in collisions. Each MR that receives a scheduling request from an adjacent MR will reply with a positive acknowledgment if it is not in the process of activating its own schedule in the current frame (line 15). However, if the MR that has received a scheduling request is currently in the process of activating its own schedule, it will reply with a negative (positive) acknowledgment if it has a lower (higher) priority than the MR sending the request (lines 9-14). The MR that receives positive acknowledgments from all of its adjacent MRs activates its schedule in the next frame (line 3-5 ). It shall also update adjacent MRs with the used channels in each slot.

**Algorithm 2**: Proactive Collision-Avoidance (PCA)

1 **if** *an MR* $a_{0,i}$ *needs to activate the multicast schedule of cell* $i$ **then**
2     It broadcasts a scheduling request packet to all adjacent MRs (those managing adjacent cells);
3     **if** *all adjacent MRs accept the request by sending a positive acknowledgment (+ACK)* **then**
4         Activate the schedule in the next frame;
5         Inform all adjacent MRs about the used channels in each slot;
6     **else**
7         Retry the activation in the next frame
8 **if** *MR* $a_{0,i}$ *receives a scheduling request from an adjacent MR* $a_{0,j}$ **then**
9     **if** $a_{0,i}$ *is trying to activate its schedule in the current frame* **then**
10         **if** *cell* $i$ *has a higher priority than cell* $j$ **then**
11             $a_{0,i}$ replies with a *negative* acknowledgment to $a_{0,j}$;
12         **else**
13             $a_{0,i}$ sends a *positive* acknowledgment to $a_{0,j}$;
14             $a_{0,i}$ aborts the schedule activation and retries in the next frame;
15     **else**
16         $a_{0,i}$ sends a *positive* acknowledgment to $a_{0,j}$;

### B. Reactive approach

Under this approach, each MR calculates the schedule of the cell it manages without taking adjacent cells into consideration. This means that collisions may occur in some time slots between adjacent cells. Therefore, a collision resolution procedure is needed. This approach allows each cell to obtain the full gain of the proposed assistance mechanism. Before discussing the details, we need to highlight some properties of the schedule that the HAMS algorithm produces.

1) *The precedence property*: the first property is the precedence relationship imposed on transmissions. This relationship resembles the fact that an assistance candidate MC cannot perform its assistance by transmitting a particular codeword unless it has already received it (or a combination that can produce it) through an earlier transmission(s). For example, transmissions $x_1{=}\{z_1, v_1, k_1, (z_3, \cdots)\}$ and $x_2{=}\{z_2, v_2, k_2, (z_3, \cdots)\}$ must precede transmission $x_3{=}\{z_3, v_1 \oplus v_2, k_3, (\cdots)\}$. We represent this precedence relationship using the following notation, $x_2{\prec}x_3$ and $x_1{\prec}x_3$. In general, $x_i{\prec}x_j$ if the codeword $v_i$ was necessary to construct the codeword $v_j$ in the original cell schedule.
2) *The conflict property*: any two multicast transmissions $x_1$ and $x_2$ cannot be scheduled in the same time slot if any of the following *collision conditions* hold.
   a) $z_1 = z_2$.
   b) $\mathcal{R}_1 \cap \mathcal{R}_2 \neq \emptyset$.
   c) $k_1 = k_2$ and $c(z_1) = c(z_2)$.
   d) $k_1{=}k_2$, $c(z_1){\neq}c(z_2)$, and $\exists r_1{\in}\mathcal{R}_1 : \|r_1, z_2\|{\leq}\zeta$ or $\exists r_2{\in}\mathcal{R}_2 : \|r_2, z_1\|{\leq}\zeta$. $\zeta$ is the interference range.

Let $\digamma(x_i, x_j)$ be the collision function defined as:

$$\digamma(x_i, x_j) = \begin{cases} 1 & \text{if any of the collision conditions} \\ & \text{mentioned earlier is satisfied} \\ 0 & \text{otherwise} \end{cases} \quad (12)$$

Given the two properties explained earlier, proposing a *distributed* algorithm that can resolve collisions without wasting the gain achieved by the assistance operation is not an easy task. Therefore, we adopt the *proactive* approach as the solution for collision resolution. Furthermore, we propose an ILP formulation to resolve the collisions for the reactive approach. The performance of this ILP will be used a baseline reference to evaluate the performance of the reactive approach. The basic idea of the ILP is to fit the schedules of $C$ cells in the shortest time frame possible, we refer to this frame as the network span, such that no collisions happen in any time slot.

The shorter the network span the better because it leads to higher throughput for the CR-WMN and smaller probability of collision with the primary network.

Given the multicast schedules $\{\mathcal{X}_1, \cdots, \mathcal{X}_C\}$ of the total $C$ cells in the network, such that $\mathcal{X}_i = \{\mathcal{X}_i[1], \cdots, \mathcal{X}_i[\tau_i]\}$ where $\tau_i$ is the length of the schedule $\mathcal{X}_i$ which is obtained using Algorithm 1. Let $\overline{\mathcal{X}}_i = \bigcup_{t=1}^{\tau_i} \mathcal{X}_i[t]$. Also, let $\tau_{max} = \max_{1\leq i \leq C} \tau_i$. Then, the ILP formulation is shown next.

Minimize $\sum_{t=1}^{\tau_{max}} t \cdot \nu^t$, subject to:

$$\omega_{j,n,t} \leq \nu^t, \qquad 1 \leq n \leq C, x_j \in \overline{\mathcal{X}}_c \quad (13)$$

$$\omega_{j,n,t} + \omega_{i,m,t} \leq 1, \qquad 1 \leq t \leq \tau_{max}, 1 \leq n, m \leq C, \; x_i \neq x_j, x_i \in \overline{\mathcal{X}}_m, x_j \in \overline{\mathcal{X}}_n, F(x_i, x_j)=1 \quad (14)$$

$$\omega_{j,n,t} \leq \sum_{\hat{t}=1}^{t-1} \omega_{i,n,\hat{t}}, \quad 1 \leq t \leq \tau_{max}, 1 \leq n \leq C, \; x_i, x_j \in \overline{\mathcal{X}}_n, x_i \prec x_j \quad (15)$$

$$\sum_{t=1}^{\tau_{max}} \omega_{j,n,t} = 1, \qquad 1 \leq n \leq C, x_j \in \overline{\mathcal{X}}_n \quad (16)$$

The objective of this ILP is to minimize the length of the network span. Constraint (14) guarantees collision free solution. Constraint (15), on other hand, maintains the precedence relationship between the multicast transmissions within the same cell. Lastly, constraint (16) guarantees that each transmission is scheduled in a time slot.

## VIII. Handling transmission failures

In a CRN, channel availability is not guaranteed for SUs. The resumption of activities by a PU in a particular slot will force the SU transmitter (MR or MC) which is scheduled to transmit in this slot to abandon its schedule and back off. There is therefore a need to rerun the algorithm in order to construct a new schedule that takes into consideration the new channels availabilities (and unavailabilities). Since rerunning the algorithm may consume some time, which may last for many cycles, especially if the objective is to find a new optimal schedule, waiting for the new schedule to be computed will waste bandwidth. We propose to use a recovery mechanism that executes in a very short time, and will deliver the packets scheduled for delivery in the current cycle, and following cycles until a new schedule is found. Therefore, there is a need to devise such a recovery mechanism, and we introduce such a mechanism in this section.

We need to distinguish here between the recovery of an MR and an MC failed transmissions. The HAMS algorithm guarantees that the MR has something to transmit in each slot of the frame. Therefore, if it fails to transmit in a particular slot, there is no way for it to recover that failed transmission without discarding some other scheduled transmission(s). Therefore, when the MR fails to transmit, and the delivery delay exceeds the maximum tolerable delay for the multicast application, rescheduling will be triggered and extra slots will be added to help the MR drain its queues, as it will be explained below.

On the other hand, the MC transmitters might have some transmission opportunities throughout the frame to use for recovery. It would be beneficial to make use of such opportunities not just to deliver the missed transmission earlier, but also to unblock any future transmissions which are depending on the reception of the codeword of the missed transmission. For example, assume that MC $z$ is supposed to transmit codeword $v = p_1 \oplus p_2$ in slot $t$ given that it has received $p_1$ in slot $t-1$ and $p_2$ in slot $t-2$. If a PU occupies the channel at $t-2$ causing the scheduled transmission to fail, both codewords $p_2$ and $p1 \oplus p_2$ will be delayed.

### A. Recovery process

As explained earlier, we need a recovery process to fix the schedule in case of any interruptions caused by PU activity. In this subsection, we propose an online recovery scheduling algorithm that monitors the dynamics of a queueing system maintained at each node and based on which calculates the recovery schedules. Before we propose the recovery algorithm, we need to illustrate the queueing strategy on which the recovery algorithm will rely. Each node, $i$, that is scheduled to transmit in at least one slot (either the MR or an assistance MC) will maintain the following queues:

- **Input queue (IQ)**: this queue holds the received codewords, and it is parameterized by the frame ID (i.e., a node $i$ maintains a queue $IQ_i[f]$ for each frame $f$).
- **Availability queue (AQ)**: this is a virtual queue that holds all combinations of the codewords in IQ and is again parameterized by the frame ID, $AQ_i[f]$. Please note that this is a virtual queue used to simplify the algorithm presentation and it is not a physical queue. It is used to indicate that all the combinations needed to construct a scheduled codeword have been received.
- **Delayed Queue (DQ)**: this is a virtual queue holding all the codewords that a transmitter was unable to transmit because they are not yet available (i.e., not present in AQ at the time of transmission). This queue is parameterized by the slot ID, $DQ_i[t]$.
- **Output Queue (OQ)**: this is a physical queue holding the codewords that are available (i.e., present in AQ) but the transmitter is unable to transmit because the channel is unavailable at the scheduled transmission time. This queue is parameterized by the slot ID, $OQ_i[t]$.

Fig. 5 illustrates the interaction between these four queues. Two more points to add:

- Whenever queue $AQ_i[f]$ is updated, the following check is performed: $\forall v \in DQ_i[t]$ $(0 \leq t \leq F)$*: if* $v \in AQ_i[f]$, *then dequeue* $v$ *from* $DQ_i[t]$ *and enqueue it into* $OQ_i[t]$, where $F$ is the frame length.
- Whenever a codeword $v$ is moved out of DQ to the OQ, all codewords in the IQ which are no longer needed to construct a codeword in DQ are removed.

Whenever an MC $i$ fails to use the slot scheduled for it to transmit, say $t$, it informs the MR via the CCC together with the size of $OQ_i[t]$ (we assume that the MR either has a separate radio for control, or uses time multiplexing). Before the beginning of the next frame, the MR calculates recovery schedules for all nodes with non-zero OQ's giving priority to the ones with the largest OQ size as outlined in Algorithm 3. The MR then informs the MCs about the calculated recovery schedule via the control channel as it will be described later. The extra transmissions scheduled for recovery purposes are valid for one frame only, and the MR will recalculate recovery schedules every frame, as needed.

The MR calculates the recovery schedule for MCs in a greedy manner as outlined in algorithm 3. The OQ with the largest size is processed first. Let that be $OQ_{i^*}[t^*]$. The MR looks up the transmission details (i.e., codeword and receivers) from the original schedule (calculated by HAMS). Then, it iterates over the slots in the frame trying to schedule a transmission that serves the maximum number of receivers and at the same time does not conflict with any scheduled transmission in the original schedule. If it succeeds to serve all the receivers of the failed transmission, it adds the found transmission opportunities to the original schedule. Otherwise, it ignores this OQ (by setting the size to zero to make it ignorable). This operation repeats until there is no more non-zero OQ's to process. The MR then sends the new calculated schedule to all MCs via the control channel (please recall that the extra scheduled transmissions are valid for one frame only). It is also possible that no recovery is possible for some transmitters. In such case, the MR will do nothing. It will just wait for a notification from a receiver MC that the packet delay has exceeded the maximum tolerable delay by the multicast application. If the MR receives such a notification, it triggers the full rescheduling (i.e., running the HAMS again) taking into consideration the avoidance of the channels which have caused consistent growth in OQ's. The MR will also add the minimum number of extra slots that the transmitters (including the MR itself) which have non-zero OQ's can share (using Algorithm 3) to drain their OQ's. If such extra slots are added, the MR will have some idle slots to use for recovering its own OQ's. Once all transmitters drain their OQ's, the MR will shrink the schedule back by removing the extra slots it added earlier, and notify the MCs about this change. Any changes made to the schedule are communicated with the adjacent cells (via the CCC) to maintain the collision free atmosphere.

To explain the recovery behavior, we present the example summarized in Fig. 4. We simulate the schedule in Fig. 4 with packet size of 1555 bytes and slot length of 2.43 $ms$ for six seconds. The simulation results are presented in Fig. 6. In period $[1, 2]$, we blocked channel 3 in slot 3 (i.e., used by a PU) only, and at the same time we disabled failure recovery. Thus, the size of $OQ_4[3]$ increased linearly until it hit 58. Then we enabled the failure recovery in $[2-3]$, while keeping channel 3 busy in slot 3 during this period. According to the schedule, node $n_4$ can make use of channel 3 in slots $\{0, 1, 5, 6\}$ for recovery. Therefore, the MR scheduled those slots to be used by $n_4$ to drain $OQ_4[3]$, and the queue quickly drained at rate 4 $packets/frame$. In period $[3-4]$, we blocked channel 3 in slot 2 and disabled failure recovery, and therefore $OQ_5[2]$ has built up. We enabled the recovery back at time $t = 4$, and blocked channel 3 in slots 0 and 1 till the end to the simulation period. According to the schedule, node $n_5$ can make use of channel 3 in slots 5 and 6. Therefore, the MR schedules those slots for $n_5$ to use. While $OQ_5[2]$ is draining at rate 2 $packets/frame$, $OQ_4[3]$ is building up because it now has an input rate that is higher than its output rate and at

**Algorithm 3**: Greedy recalculation of recovery schedules

**input** : $OQ_i[t],\ \forall i \in \mathcal{Z},\ 0 \le t < F$;
$\mathcal{Z}$: the set of transmitters in the permanent schedule of the next frame;
$F$: the frame length;
**output**: $\mathcal{X}_{total}$: The schedule of the next frame including temporary recovery transmissions;

Copy the permanent schedule of the next frame into $\mathcal{X}_{total}$;
**while** $\exists(i, t) : OQ_i[t] > 0$ **do**
  Let $OQ_{i^*}[t^*]$ be the OQ with the largest size;
  From the permanent schedule, look up the scheduled transmission at slot $t^*$. Let that be $x^* = \{i^*, v, k, \mathcal{R}\}$;
  $\mathcal{X} \leftarrow \emptyset$;
  **for** $\tau = 0; \tau < F; \tau++$ **do**
    **if** $|\mathcal{R}| = 0$ **then**
      **break**;
    **if** $i^*$ *and at least one receiver in* $\mathcal{R}$ *are idle in* $\mathcal{X}_{total}[\tau]$ *and share a common idle channel* **then**
      Let $k^* \in \mathcal{L}_{i^*}$ be an idle channel in $\tau$ that is available to the maximum number of receivers represented by the set $\mathcal{R}^* \subseteq \mathcal{R}$;
      $\mathcal{X} \leftarrow \mathcal{X} \cup (i^*, v, k^*, \mathcal{R}^*)$;
      $\mathcal{R} \leftarrow \mathcal{R} \backslash \mathcal{R}^*$;
  **if** $|\mathcal{R}^*| < |\mathcal{R}|$ **then**
    $OQ_{i^*}[t] = 0$
  **else**
    $\mathcal{X}_{total} \leftarrow \mathcal{X}_{total} \cup \{x\},\ \forall x \in \mathcal{X}$
**return** $\mathcal{X}_{total}$;

the same time is unable to win any recovery slots because the size of $OQ_5[2]$ is still higher. When the sizes of $OQ_4[3]$ and $OQ_5[2]$ became equal, the MR started to make slots 5 and 6 shareable between $n_4$ and $n_5$ for recovery and the recovery rate became 1 packet/frame/node until time 4.7. At $t = 4.7$, we blocked channel 3 in slot 5 leaving only one slot available for recovery purposed. Therefore, the recovery rate dropped down to 1 packet/frame until the OQ's of nodes $n_4$ and $n_5$ completely drained at $t = 5.4$. Also, note that the size of $DQ_4[3]$ matches that of $OQ_5[2]$ as expected.

## IX. Performance Evaluation

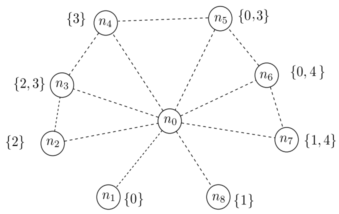
In this section, we evaluate the performance of the proposed assistance and collision resolution mechanisms. To evaluate the performance of the HAMS algorithm, we implemented it using the C# programming language. Different network topologies were generated and fed to this program to evaluate the gain of the HAMS algorithm. Also, we developed an in-house Java simulator to evaluate the schedule recovery and collision avoidance algorithms between adjacent cells.

Before presenting the results, we illustrate the channel availability models we used in our simulations. In this section, $M$ will denote the number of multicast sessions, i.e., $M = |\mathcal{S}|$.

### A. Channel Availability Models

In this subsection, we outline a number of possible channel availability models in CRNs. However, we only use the third, i.e., CM, model for our simulations.

1) *The spatial model (SM)*: under this model, a number of PUs, $N_{PU}$, are uniformly distributed in the network




| Slot | Transmitter | Channel | Codeword | Receivers |
|---|---|---|---|---|
| 0 | $n_0$ | 0 | $a$ | $\{n_1, n_5, n_6\}$ |
| 1 | $n_0$ | 0 | $b$ | $\{n_1, n_5, n_6\}$ |
| 2 | $n_5$ | 3 | $a$ | $\{n_4\}$ |
| 2 | $n_0$ | 1 | $a$ | $\{n_7, n_8\}$ |
| 3 | $n_4$ | 3 | $a$ | $\{n_3\}$ |
| 3 | $n_0$ | 1 | $b$ | $\{n_7, n_8\}$ |
| 4 | $n_0$ | 3 | $a \oplus b$ | $\{n_3, n_4\}$ |
| 5 | $n_0$ | 2 | $a$ | $\{n_2\}$ |
| 5 | $n_0$ | 2 | $b$ | $\{n_2\}$ |

Fig. 4. A case study to illustrate the recovery processes. The figure to the left shows the network topology and channel availability, while the table to the right shows the calculated schedule.

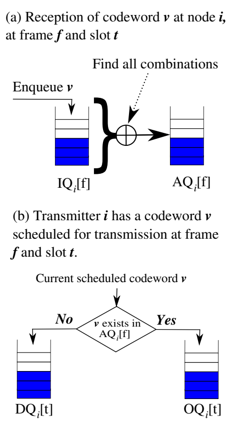



Fig. 5. Interaction between different queues.

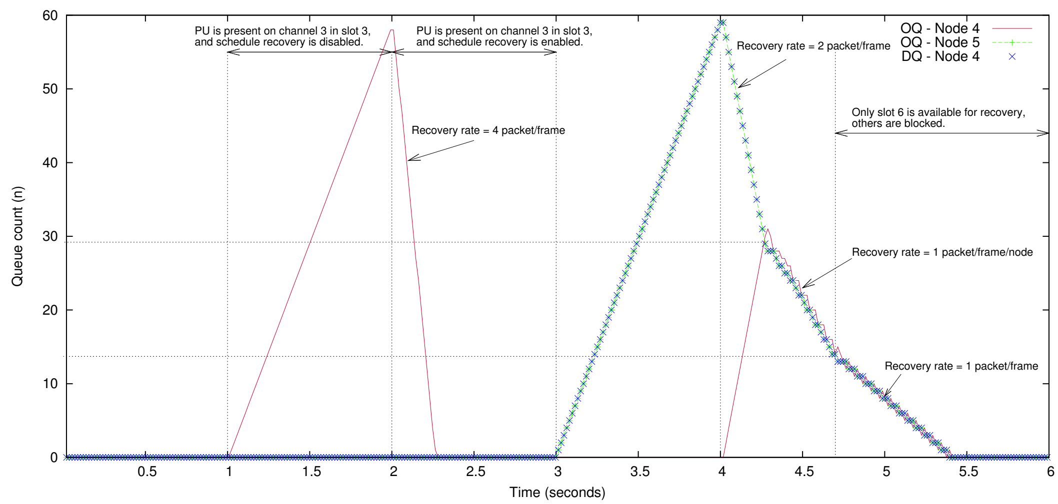
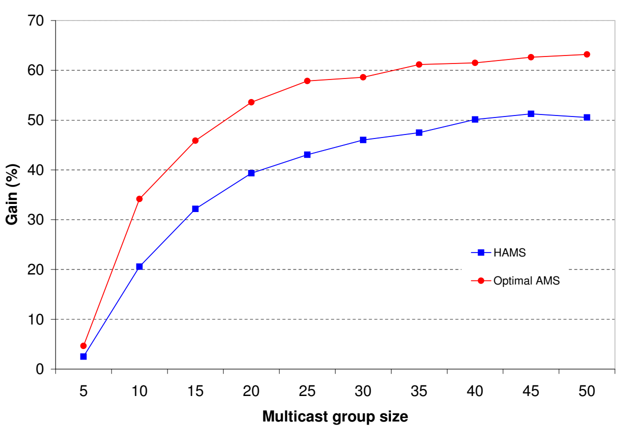
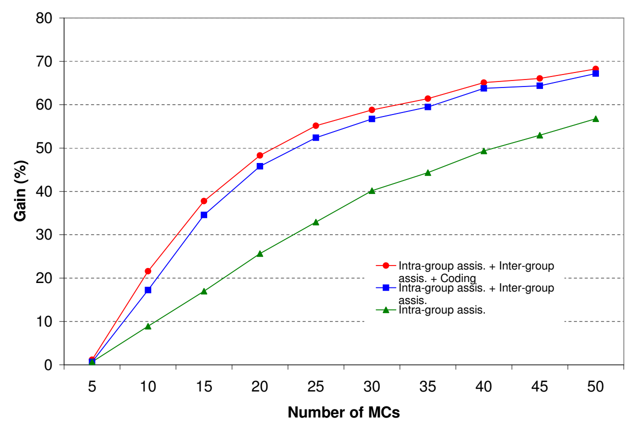



Fig. 6. The queue dynamics of the case-study summarized in Figure 4.



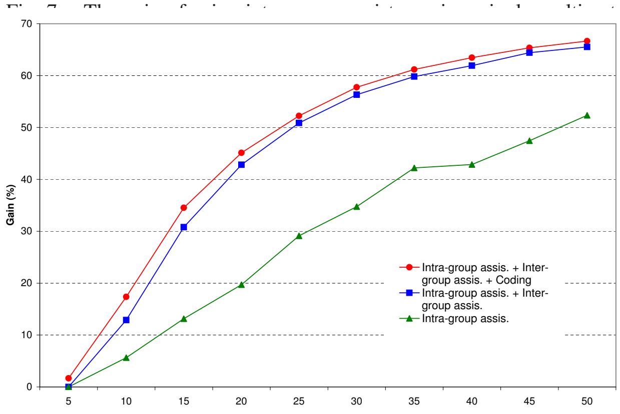




Fig. 9. Average gain of assisted multicast using different levels of assistance ($M = 4, Pa = 0.25$).

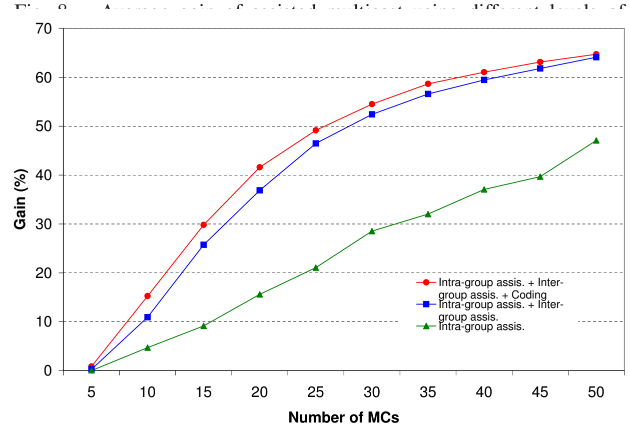



Fig. 10. Average gain of assisted multicast using different levels of assistance ($M = 5, Pa = 0.25$).

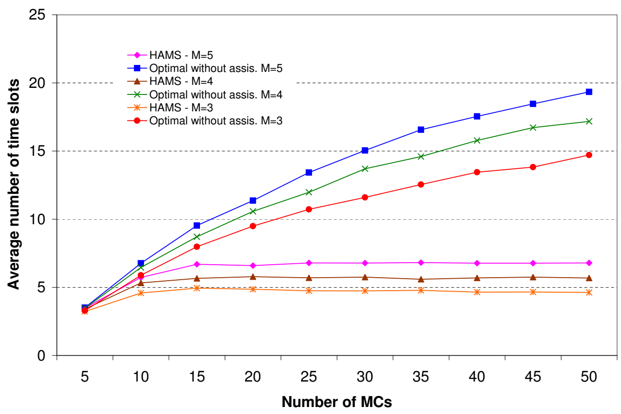



Fig. 11. Average multicast period with- and without-assistance ($M$=3, 4, 5, $Pa$=0.25).

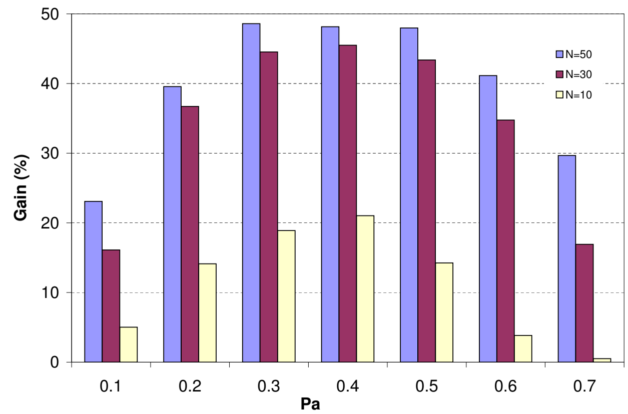



Fig. 12. The effect of channel availability on the gain of assisted multicast (M=1).

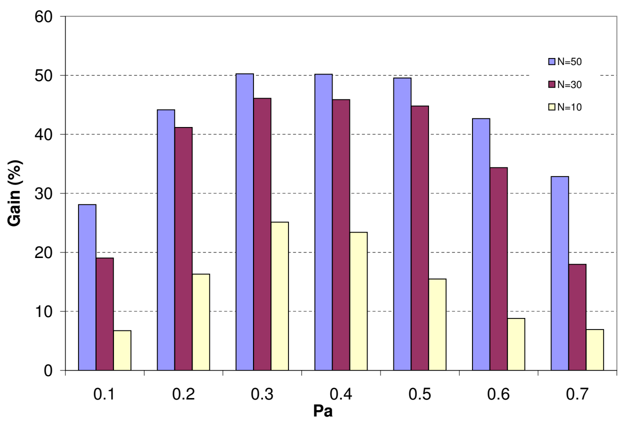



Fig. 13. The effect of channel availability on the gain of assisted multicast (M=5).

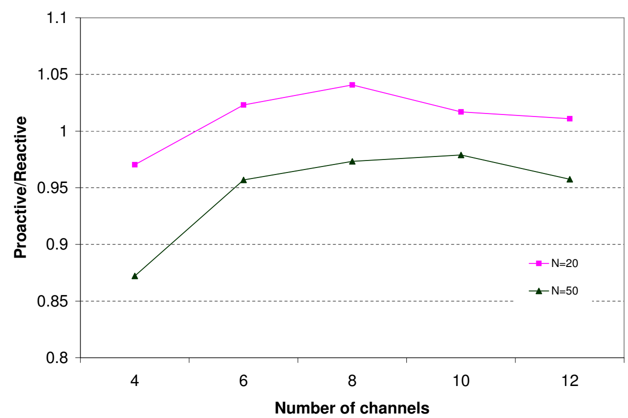



Fig. 14. Proactive versus reactive collision resolution.

field. Each PU is assigned one frequency channel selected uniformly at random from a pool of $K$ channels. An SU $j$ can use a frequency channel $k$ *iff* all PUs assigned channel $k$ are at least $R_p$ away from $j$.

2) *The hybrid model (HM)*: as in the spatial model, a number of PUs, $N_{PU}$, are uniformly distributed in the network field. Each PU is assigned one frequency channel selected uniformly at random from a pool of $K$ channels. A PU is active (i.e., using the assigned channel) with probability $p_{ON}$, and inactive with probability $1 - p_{ON}$. An SU $j$ can use a channel $k$ *iff* all *active* PUs assigned channel $k$ are at least $R_p$ away from $j$.

3) *The coexistence model (CM)*: each MR serves both primary and secondary clients, which means that MRs can utilize all frequency channels. SUs are treated as second class users which receive best effort service. Therefore, the set of available channels at SUs will depend on many different factors including, spatial distribution of PUs and SUs, primary traffic loads, load balancing between frequency channels, etc. To keep the model simple, we make all channels available to all MRs, while a channel is made available to an MC with probability $P_a$.

### B. The gain of receiver assistance

To evaluate the gain of the proposed assistance mechanism, we study a single cell with the number of MCs varying from 5 to 50. The MCs are distributed uniformly at random in a square area of $500m \times 500m$ around the MR which is located in the center of the square area. All nodes (MCs and the MR) are assumed to have the same communication radius of $\sqrt{2} \times \frac{500}{2} = 353.55m$ over all channels. We vary the number of multicast groups $M$ between 1, 3, 4, and 5. Each MC is assigned to any of the $M$ groups uniformly at random, i.e., each MC belongs to exactly one multicast group. Lastly, we have the number of channels $K = 6$ in all experiments. Available channels are determined at each node (MC or MR) according to the CM model. Using the aforementioned settings, we generated random topologies which we then fed to the C# program to identify the gain of each operation.

*a) Intra-group assistance:* Fig. 7 shows the gain of using intra-group assistance in a single multicast group. The gain is defined as the percentage reduction in the multicast period of the unassisted multicast achieved by using assisted multicast ($\frac{unassisted - assisted}{unassisted} \times 100\%$). The optimal solutions for the two cases of unassisted multicast and intra-group assisted multicast were obtained using the two ILPs proposed in Section V. We also evaluated the gain of intra-group assisted multicast by scheduling the problem using the HAMS algorithm. Each point is the figure is the average over a 100 randomly generated topologies. As the figure shows, the intra-group assistance achieves a significant gain over the unassisted case that increases with increasing the group size. On the other hand, the HAMS algorithm is performing well by achieving a considerable gain and being always within, on average, one time slot of the optimal solution obtained by

ILP formulation of the AMS-Single problem. In fact, HAMS was, on average, $\approx 0.63$ slots higher than the optimal assisted multicast schedule, and $\approx 2.11$ slots less than the optimal unassisted multicast schedule.

*b) Inter-group assistance:* We now evaluate the benefit of using each of the three assistance operations: *intra-group assistance*, *inter-group assistance*, and the *codeword exchange operation* for multiple multicast groups. We vary the number of groups $M$ between 3, 4 and 5. For each case, we evaluate the gain using *intra-group assistance only*, *intra- and inter-group assistance*, and *intra- and inter-group assistance with network coding*. For the unassisted multicast case, we find the optimal schedule for each one of the $M$ groups and summing up the optimal multicast periods for all individual groups to obtain the total multicast period. As for the assisted multicast scheduling, we used the HAMS algorithm. Figures 8, 9, and 10 correspond to the cases of $M = 3, 4$, and 5 respectively with each point in the figure be the average over a 100 randomly generated topologies. As the figures show, each level of assistance achieves some extra gain in the total multicast period. However, it is apparent that inter-group assistance has more influence on the total gain than the codeword exchange operation, yet the codeword exchange operation can still improve the scheduling performance. Fig. 11 shows the actual averages of the multicast period for the data presented in Figures 8, 9, and 10.

It is to be noted that network coding is always the last step, and can only be used with inter-group assistance. And, inter-group assistance is used after individual frames from separate group are transmitted, which enables the use of intra-group assistance. Therefore, most of the gain is achieved first through intra-group assistance. Some additional gain, which is close to 50% of the former gain, can still be achievable through inter-group assistance. Since most of the feasible gain has already been achieved, network coding adds some more gain, which is less than that introduced by the first two.

*c) The effect of channel availability:* To understand the effect of channel availability on the achievable gain of the assisted multicast, we varied $P_a$ from $0.1$ to $0.7$ for the cases of $M = 1$ and $M = 5$ as shown in Figures 12 and 13 respectively. The number of MCs in a single cell is varied between 10, 30, and 50. All MCs are assumed to be members of all multicast groups to nullify the effect of diversity in group membership on the achievable gain. Each point on the curve of any of the two figures is the average of 200 randomly generated instances. As the two figures show, the gain increases as $P_a$ increase until reaching a peak and then starts decreasing. The $P_a$ at which the gain is maximized offers the highest level of diversity in the network, the basic property on which the proposed assistance mechanism relies. Another thing to note is that the gain is higher with higher values of $M$.

### C. Proactive vs. reactive collision resolution

In this section, we study a network of 9 cells, all of which have the same number of MCs and share the same pool of channels. The number of channels is chosen from the set $\{4, 6, 8, 10, 12\}$, while the number of multicast groups is set to 3. All MCs are members of all multicast groups to nullify the effect of diversity in group membership on the achievable gain. The number of MCs in each cell is chosen from the set $\{20, 50\}$. In each experiment, all the cells have the same number of MCs. The number of cells in the network is 9, arranged in a grid of $3 \times 3$ in a field of area $500m \times 500m$. The communication radius for all nodes is $\sqrt{2} \times \frac{500}{6}$, and $\zeta$ is twice the communication radius. Using the simulation setting earlier, different randomized scenarios were generated and fed to the Java in-house simulator to simulate the collision resolution process. For each generated network scenario, each cell calculates its multicast schedule using the HAMS algorithm. Then, we use the proactive (the PCA algorithm) or the reactive (the ILP) to resolve collisions between adjacent cells. Fig. 14 shows the performance of the two approaches represented by the ratio of the proactive approach to the reactive approach for both 20 and 50 MCs in each cell. Each point on the curve is the average of 100 randomly generated instances. As the figure indicates, the performance of the proactive approach is close to that of the reactive approach with optimal collision resolution (less than 5% difference). The figure also shows that for small number of channels (4 for example), the proactive performs better than the reactive approach. Therefore, given the simplicity of implementing the proactive approach (compared to the reactive), and the good performance the figure implies, we adopt the proactive PCA algorithm to resolve collisions.

## X. CONCLUSIONS

In this work, we have studied the problem of assisted multicast scheduling in wireless cognitive mesh networks. We have proposed an assistance paradigm that relies on receiver nodes to forward the multicast data to other receivers that have not yet received their own data. Furthermore, network coding was also proposed as another assistance technique that further reduced the total multicast period. Results show that the proposed assistance paradigm achieves a significant gain in reducing the total multicast period, i.e., overall throughput. A proactive collision resolution procedure was also proposed to build collision-free schedules across cells in a CR-WMN.

Future research includes considering the implementation of our proposed approaches using standard MAC protocols, or developing new MAC protocols to support this approach, while still being compatible with other standard protocols.

## REFERENCES


[1] Joseph Mitola. *An Integrated Agent Architecture for Software Defined Radio*. PhD thesis, Royal Institute of Technology (KTH), Stockholm, Sweden, May 2000.

[2] Ian F. Akyildiz et al. Next generation/dynamic spectrum access/cognitive radio wireless networks: a survey. *Computer Networks*, 50(13):2127–2159, 2006.

[3] K.R. Chowdhury and I.F. Akyildiz. Cognitive wireless mesh networks with dynamic spectrum access. *Selected Areas in Communications, IEEE Journal on*, 26(1):168–181, January 2008.

[4] Christina Fragouli et al. Network coding: an instant primer. *SIGCOMM Comput. Commun. Rev.*, 36(1):63–68, 2006.

[5] Ivana Maric and Roy D. Yates. Cooperative multicast for maximum network lifetime. *IEEE JSAC*, 23(1):127 – 135, 2005.

[6] Tehuang Liu and Wanjiun Liao. Multicast routing in multi-radio multi-channel wireless mesh networks. *IEEE Transactions on Wireless Communications*, 9(10):3031 –3039, October 2010.

[7] Hon Sun Chin et al. Bandwidth-guaranteed multicast in multi-channel multi-interface wireless mesh networks. In *IEEE International Conference on Communications (ICC)*, pages 1 –5, June 2009.
[8] Nguyen Hoang Lan and Nguyen Uyen Trang. Channel assignment for multicast in multi-channel multi-radio wireless mesh networks. *Wireless Communications and Mobile Computing*, 9:557–571, April 2009.
[9] Guokai Zeng et al. Multicast algorithms for multi-channel wireless mesh networks. In *IEEE International Conference on Network Protocols*, pages 1–10, October 2007.
[10] Guokai Zeng et al. Efficient multicast algorithms for multichannel wireless mesh networks. *IEEE Transactions on Parallel and Distributed Systems*, 21:86–99, 2010.
[11] V. Ramamurthi et al. Multicast capacity of multi-channel multihop wireless networks. In *IEEE Wireless Communications and Networking Conference*, pages 1 –6, April 2009.
[12] Ayaz Isazadeh and Mohsen Heydarian. Traffic distribution for end-to-end qos routing with multicast multichannel services. *The Journal of Supercomputing*, 52:47–81, 2010.
[13] Miao Ma and D.H.K. Tsang. Impact of channel heterogeneity on spectrum sharing in cognitive radio networks. In *IEEE International Conference on Communications (ICC)*, pages 2377 –2382, May 2008.
[14] Mark A. McHenry et al. Chicago spectrum occupancy measurements & analysis and a long-term studies proposal. In *Proceedings of the first international workshop on Technology and policy for accessing spectrum*, TAPAS '06, New York, NY, USA, 2006. ACM.
[15] Srikanth Pagadarai and Alexander M. Wyglinski. Measuring and modeling spectrum occupancy: A massachusetts perspective. In *Proceedings of the 2010 International Symposium on Advance Radio Technologies*, July 2010.
[16] Sixing Yin et al. Mining spectrum usage data: A large-scale spectrum measurement study. *IEEE Transactions on Mobile Computing*, June 2012.
[17] Manuj Sharma and Anirudha Sahoo. Residual white space distribution-based opportunistic channel access for cognitive radio enabled devices. *SIGCOMM Comput. Commun. Rev.*, 41(4), August 2010.
[18] Ahmet Akbulut et al. Estimation of time-varying channel state transition probabilities for cognitive radio systems by means of particle swarm optimization. *RadioEngineering*, 21, April 2012.
[19] Duy T. Ngo et al. Resource allocation for ofdm-based cognitive radio multicast networks. In *IEEE Wireless Communications and Networking Conference (WCNC)*, 2009.
[20] Jin Jin et al. Multicast scheduling with cooperation and network coding in cognitive radio networks. In *IEEE INFOCOM*, 2010.
[21] W. Kim, S. Oh, M. Gerla, and J.-S. Park. Cocast: Multicast mobile ad hoc networks using cognitive radio. *IEEE MILCOM*, 2009.
[22] H. M. Almasaeid, T. H. Jawadwala, and A. E. Kamal. On demand multicast routing in cognitive radio mesh networks. In *IEEE Global Communications Conference (GLOBECOM)*, December 2010.
[23] W. Ren et al. Minimum-energy multicast tree in cognitive radio networks. *IEEE Asilomar Conference*, 2009.
[24] C. Xin et al. A novel layered graph model for topology formation and routing in dynamic spectrum access networks. In *IEEE Int. Symp. on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, 2005.
[25] Sung Ju Lee, William Su, and Mario Gerla. On-demand multicast routing protocol in multihop wireless mobile networks. *Mobile Network Applications*, 7:441–453, December 2002.
[26] R. Ahlswede et al. Network information flow. *IEEE Transactions on Information Theory*, 46(4):1204 –1216, July 2000.
[27] D.S. Lun et al. Achieving minimum-cost multicast: a decentralized approach based on network coding. In *IEEE INFOCOM*, volume 3, pages 1607–1617, March 2005.
[28] T. Ho, M. Medard, R. Koetter, D.R. Karger, M. Effros, Jun Shi, and B. Leong. A random linear network coding approach to multicast. *IEEE Transactions on Information Theory*, 52(10):4413 –4430, October 2006.
[29] O.M. Al-Kofahi and A.E. Kamal. Network coding-based protection of many-to-one flow networks. In *IEEE Internatonal Conference on Mobile Ad hoc and Sensor Systems (MASS)*, pages 1–10, October 2007.
[30] O. M. Al-Kofahi and A. E. Kamal. Network coding-based protection of many-to-one wireless flows. *IEEE J-SAC special issue on Network Coding in Wireless Communications*, 27(5):797–813, 2009.
[31] Lijuan Geng, Ying-Chang Liang, and F. Chin. Network coding for wireless ad hoc cognitive radio networks. *IEEE PIMRC*, pages 1–5, September 2007.
[32] C.-H. Huang, Y.-C. Lai, and K.-C. Chen. Network capacity of cognitive radio relay network. *Physical Communication*, 1:112–120, June 2008.
[33] Sachin Katti et al. Embracing wireless interference: analog network coding. In *SIGCOMM*, pages 397–408. ACM, 2007.
[34] X. Chen et al. Inter-cluster connection in cognitive wireless mesh networks based on intelligent network coding. In *EURASIP Journal on Advances in Signal Processing*, 2009.
[35] V. Fodor, I. Glaropoulos, and L. Pescosolido. Detecting low-power primary signals via distributed sensing to support opportunistic spectrum access. *IEEE International Conference on Communications (ICC)*, 2009.
[36] SENDORA project. http://www.sendora.eu.
[37] B. Mercier, V. Fodor, et al. Sensor networks for cognitive radio: Theory and system design. *ICT Mobile Summit*, June 2008.

## APPENDIX A
## PROOF OF THEOREM 5.1

*Proof:* A reduction from the set-cover problem can be easily drawn. The set cover problem has, as input, a universe $\mathcal{U}$ and a group of subsets $\mathcal{S} = \{S_1, \cdots, S_M\}$, and the objective is to find the minimum number of subsets that cover the universe $\mathcal{U}$, i.e., Minimize $|\mathcal{C}| : \mathcal{C} \subseteq \mathcal{S}, \ \bigcup_{c \in \mathcal{C}} c = \mathcal{U}$. To map an instance of the set-cover problem into an instance of the UMS-single, we do the following:

- Create a hypothetical node $n$ and mark it as the MR.
- For each member $u \in \mathcal{U}$ in the set-cover problem, create an MC $u$ in the UMS-Single problem and extend an edge between $u$ and $n$.
- Map each subset $S_k$ in the set-cover problem into a channel $k$ in UMS-Single problem. Then, make channel $k$ available to every MC $u$ *iff* $u \in S_k$.
- Make all channels available to the MR $n$.

Note that in the UMS-Single problem, MR $n$ is the only transmitter and it transmits on one channel at each time slot. Also, note that any solution that has the MR transmits on the same channel in different time slots is not optimal, because the exact same set of MCs will receive the packet in both transmissions. Therefore, the minimum number of time slots to deliver the multicast packet to all MCs maps directly, by construction, to the minimum number of sets that can cover $\mathcal{U}$. In the other direction, the minimum number of subsets that cover the universe maps, also by construction, to the minimum number of time slots needed to deliver the multicast packet.

■