Wireless Netw (2011) 17:411–435
DOI 10.1007/s11276-010-0288-5

A non-cooperative game-theoretic approach to channel assignment in multi-channel multi-radio wireless networks

Rohith Dwarakanath Vallam · Arun A. Kanagasabapathy · C. Siva Ram Murthy

Published online: 11 November 2010

Abstract Channel assignment in multi-channel multi-radio wireless networks poses a significant challenge due to scarcity of number of channels available in the wireless spectrum. Further, additional care has to be taken to consider the interference characteristics of the nodes in the network especially when nodes are in different collision domains. This work views the problem of channel assignment in multi-channel multi-radio networks with multiple collision domains as a non-cooperative game where the objective of the players is to maximize their individual utility by minimizing its interference. Necessary and sufficient conditions are derived for the channel assignment to be a Nash Equilibrium (NE) and efficiency of the NE is analyzed by deriving the lower bound of the price of anarchy of this game. A new fairness measure in multiple collision domain context is proposed and necessary and sufficient conditions for NE outcomes to be fair are derived. The equilibrium conditions are then applied to solve the channel assignment problem by proposing three algorithms, based on perfect/imperfect information, which rely on explicit communication between the players for arriving at an NE. A no-regret learning algorithm known as Freund and Schapire Informed algorithm, which has an additional advantage of low overhead in terms of information exchange, is proposed and its convergence to the stabilizing outcomes is studied. New performance metrics are proposed and extensive simulations are done using Matlab to obtain a thorough understanding of the performance of these algorithms on various topologies with respect to these metrics. It was observed that the algorithms proposed were able to achieve good convergence to NE resulting in efficient channel assignment strategies.

Keywords Multi-channel multi-radio wireless networks · Channel assignment · Centralized and distributed algorithms · Game theory

1 Introduction

The dawn of the age of ubiquitous wireless networks has given rise to myriad issues related to growing bandwidth demand coupled with scarcity of available spectrum. These challenges make wireless networks a very exciting area of research. Significant advances by the research community in the direction of bandwidth enhancement in wireless networks has enabled the ability to utilize multiple channels within the same neighbourhood, thus resulting in increasing the effective bandwidth available to wireless network nodes. For example, the IEEE 802.11b/802.11g standards [1] and IEEE 802.11a standard [2] provide three and twelve non-overlapping frequency channels respectively. Further, single radio interface per node in a multi-channel wireless network resulted in the overhead of channel-switching at the medium access (MAC) layer. Another significant milestone related to bandwidth aggregation resulted through the support for multiple radio interfaces in each wireless node. Today, there is a lot of focus on performance of these multi-channel multi-radio

R. D. Vallam (✉)
Indian Institute of Science (IISc), Bengaluru, India
e-mail: rohithdv@csa.iisc.ernet.in

A. A. Kanagasabapathy
Bloomberg, L.P., London, UK
e-mail: akanagasaba1@bloomberg.net

C. S. R. Murthy
Department of CSE, IITM, Chennai, India
e-mail: murthy@iitm.ac.in

(MCMR) networks. There have been new architectures of MCMR networks proposed to cater to additional demands like self-configurability, flexibility and easy deployment. MCMR networks promise several advantages such as low deployment cost, easy maintenance and reliable service coverage. There is also a need to support bandwidth-intensive applications such as streaming video, real-time voice and video traffic with better Quality of Service (QoS) guarantees in such networks. Such guarantees are highly dependent on the *channel assignment* algorithm which effectively utilizes and distributes the network resources. Also, due to the usage of multiple radio interfaces, the throughput guarantees provided by these networks are very high. Thus, in order to utilize the multiple radios and multiple channels effectively, efficient channel assignment algorithms are needed.

This paper addresses the problem of channel assignment in MCMR wireless networks. Further, this issue is addressed in *multiple collision domain* (MCD) context i.e., each communication link of the network interferes with a subset of all the links of the network.

The main contributions of this paper can be put forth as follows.

- The problem of channel assignment in an MCMR wireless network with multiple collision domains is considered and modelled as a non-cooperative game where individual players exhibit rational behaviour and aim to maximize their individual throughput. Necessary and sufficiency conditions for this game to have the Nash equilibrium (NE) are derived. The efficiency of the NE is analyzed through establishing lower bound on the price of anarchy (PoA). QoS aspects of the problem are studied by proposing a new measure for fairness in MCMR-MCD game and necessary and sufficient conditions for NE outcomes to be fair are derived.
- Four channel assignment algorithms based on the NE conditions are proposed. The first is a centralized algorithm with perfect information about all the channels, the second algorithm also has perfect information on all the channels but is in a distributed scenario, and the third is another distributed algorithm but with imperfect information on interfering radios in the channels. The fourth algorithm is a no-regret learning-based distributed algorithm applied in a uninformed setting where every player need not have knowledge of other players.
- Finally, performance evaluation metrics based on convergence are defined and simulations have been performed to evaluate these algorithms based on these metrics.

The rest of the paper is organized as follows. Section 2 discusses related work in the area of channel assignment. Section 3 explains the problem setting by introducing the system model through the conflict graph representation and analytically formulates the channel assignment problem as a non-cooperative game. The concept of NE and analytical conditions on NE for MCMR networks in a MCD scenario along with efficiency analysis are presented in Sect. 4. Fairness analysis of the problem is presented in Sect. 5. We propose four algorithms in Sect. 6 to solve the channel assignment problem, each in a different setting. Finally, Sect. 7 presents detailed performance evaluation results of the algorithms proposed. Section 8 discusses the conclusions and future work.

2 Related work and motivation

Channel assignment in wireless networks has been an interesting topic that has been actively discussed by the research community. A large number of solutions have been suggested for cellular networks and wireless local area networks (WLANs). Techniques such as graph coloring, neural networks, simulated annealing and tabu search have been applied in solving the channel assignment problem. A comprehensive review on various solutions for frequency assignment in cellular networks is given in Aardal et al. [3]. Riihijarvi et al. [4], Leung and Kim [5] propose solutions in a centralized as well as distributed setting to the channel assignment problem in WLANs. Mishra et al. [6] studied channel assignment as a weighted graph coloring problem in WLANs.

Alicherry et al. [7] jointly considered routing problem with channel assignment. They present linear programming (LP) formulation to maximize the aggregate throughput. They also propose a centralized algorithm that assigns channels to node radios and finds routing paths. Shin et al. proposed a distributed channel assignment algorithm in [8] which uses a skeleton assisted channel assignment strategy. It relies on the construction of a spanning subgraph (called skeleton) of the connectivity graph using a distributed algorithm Local Minimum Spanning Tree (LMST) [9]. The proposed protocol uses the constructed skeleton to assign channels while preserving connectivity. Das et al. considered a fixed MCMR network in [10] and propose a mixed integer linear program (MILP) based static channel assignment scheme that maximizes the number of bidirectional links that can be activated simultaneously subject to interference constraints.

Most of the approaches used in channel assignment discussed so far are based on the inherent premise that there is cooperation among nodes in order to achieve high throughput from the entire network point of view. However, realistic scenarios show that nodes will be *selfish* by nature and will want to maximize their own throughput regardless of the performance of the network. It is in these contexts that game theoretic approaches have found tremendous applicability. Applying game-theoretical tools have been found successful in modelling various problems in wireless networks where issues of conflict and cooperation arise. Cagalj et al. [11] and Konorski [12] investigated the behaviour of CSMA/CA protocol amidst selfish users who can violate the protocol deferment times to gain unfair share of bandwidth. MacKenzie and Wicker [13] studied NE scenarios in presence of selfish users using the Aloha protocol.

Recently, game theoretical approaches have been applied to solve the channel assignment problem. Halldorsson et al. [14] analyzed the fixed channel assignment problem in a game-theoretic context. Here, the work identifies NE outcomes with the solutions to a maximal colouring problem in an appropriate graph. It also analyzes the price of anarchy and provides bounds for it and further, relate the price of anarchy of the spectrum sharing game to the approximation factor of local optimization algorithms for the maximum k-colourable subgraph problem. But the main drawback of this work is that it did not consider the scenario of multiple radio devices.

In an MCMR network, there may be interference between different transmitting links. If all the links are in the *sensing range* of each other, then the network is said to be in *single* collision domain. In a single collision domain, if all the devices are using the same channel then every device can hear the transmission of every other device. Wu et al. [15] address the problem of channel assignment by devising a payment formula so that the channel assignment converges to a strongly dominant NE instead of an NE which the authors show through experiments that it leads to higher throughput than normal NE solutions. Though it is an interesting and relevant work in channel assignment in wireless network of selfish nodes, the setup of the paper is in a single collision domain and the work does not address the issues present in the multiple collision domain scenario. Nie and Comaniciu [16] propose a game theoretic framework to analyze the behavior of cognitive radios for distributed adaptive spectrum allocation, but their main results are for cooperative users only.

The work of Felegyazi et al. [17] solved the channel assignment problem by modelling it as a non-cooperative game for single-hop MCMR networks but considered nodes only in a single collision domain. Gao and Wang [18] addressed the problem for multi-hop ad hoc wireless networks (AWNs) from a game theoretical perspective but the work was again limited to a single collision domain. However, in realistic scenarios, every device need not be in the sensing range of every other device in the network. Hence, in such a scenario, each transmitting link will have a subset of links in the network with which its interference range overlaps. In this work, we continue the investigation of the static channel assignment problem as discussed in [17, 18] and focus on solving it in a single-hop MCMR network where nodes need not have same collision domain with respect to each other (i.e., in a MCD scenario).

3 Problem setting and formulation

3.1 Generic network model

Consider a generic network, $\Lambda = (V, T)$, represented as graph where V is the number of nodes in the network and T represents the edges of the network. An edge exists between two nodes in the network if and only if there is a *communication session* in progress between them. T denotes the set of communication sessions in the network represented by $T = \{e_i = (s_i, d_i) | s_i \in S, d_i \in D \text{ and } i \in \{1, 2, \ldots, m\}\}$ where $S = \{s_1, s_2, \ldots, s_m\}$ represents the set of all source nodes in Λ and $D = \{d_1, d_2, \ldots, d_m\}$ represents set of all destination nodes in Λ. Note that $|T|$ represents the cardinality of set T and hence, $|T| = m$. We now put forth the details of our problem setting which is similar to the related works in [17, 18] but with the relaxation of the assumption about single collision domain. The single collision domain constraint is very restrictive in the sense that it requires all the communication sessions to be within a particular geographical neighbourhood so that they will *all* be in the sensing range of each other. However, in our setting, communication sessions may no longer need to be constrained to a single collision domain context and may now be able to have their own collision domains based on their geographical location. We assume each node consists of a radio device with k radio transceivers. Each radio transceiver in a device can be tuned to a particular channel for transmission or reception of data. Each source node, $s_i \in S$, has always some packets to exchange and participates in a single communication session. The wireless spectrum is divided into p channels denoted by $C = \{c_1, c_2, \ldots, c_p\}$, each of which can be used for transmission. Also, no two radios of the same node are assigned the same channel and the number of radios per node is less than the total number of channels in the wireless spectrum i.e., $k < |C|$. We consider a network setting where any node in the network should be participating in some communication session i.e., $V = S \cup D$ represents all the nodes in Λ. We assume that all the communication links are *single hop* by nature i.e., for a communication session $(s_i, d_i) \in T$, the nodes s_i and d_i are in transmission range of each other.

The channels are orthogonal and the characteristics of all the channels are identical. All the communication sessions are symmetric links i.e., the sender and receiver have same transmission ranges and thus, are able to coordinate and tune their radios to a common channel to ensure successful transmission. In an MCMR-MCD channel assignment problem, the objective is to devise a channel assignment scheme where each radio of every *transmitting* node is assigned a particular channel from the wireless spectrum which it can use for transmission of data. It will be assumed that the corresponding receiver will tune to the same channel as transmitter for reception of data. Henceforth, we will consider only transmitting nodes for all our discussion on channel assignment.

3.2 An example MCMR-MCD network

A simple example of this type of network setting is given in Fig. 1. In this figure, $S = \{s_i \mid i \in \{1, 2, 3\}\}$ and $D = \{d_i \mid i \in \{1, 2, 3\}\}$. We can see that there are three communication sessions (or links) represented by $T = \{e_i = (s_i, d_i) \mid i \in \{1, 2, 3\}\}$. From the figure, we can infer that e_1 and e_3 interfere with link e_2 independently, while link e_2 interferes with both e_1 and e_3. So, e_1, e_2, and e_3 have different interference neighbourhoods and so they are in different collision domains with respect to each other.

Throughout this paper, we will consider such type of *multiple collision domain* networks and will focus on developing a channel assignment scheme which can be applied to such generic interference aware topologies.

3.3 Conflict graph representation

We will use the conflict graph representation which is widely used in the literature (for example, Gupta et al. [19], Luo et al. [20], Jain et al. [21]) to depict the interference relation between different links in Λ. Every node in a conflict graph basically corresponds to a unique communication session in Λ. A conflict graph is represented by $\Psi = (N_{CG}, E_{CG})$. Let $N_{CG} = \{1, 2, \ldots, N\}$ denote the set of nodes in Ψ where N denotes the cardinality of the set N_{CG}. An edge $e_{ij} \in E_{CG}$ if and only if there exists two communication sessions, $(s_i, d_i), (s_j, d_j) \in \Lambda$, which *interfere* with each other. From the above construction, we can see that $N = |T|$.

We represent the interference characteristics of the given generic network, Λ, by defining an $N \times N$ Link Interference Matrix (LIM) for the conflict graph (Ψ) to capture the interference between any two communication sessions in the given network i.e., $\forall i, j \in \{1, 2, \ldots, N\}$,

$$\mathrm{LIM}(i,j) = \begin{cases} 1 & \text{if } e_{ij} \text{ is an edge in } \Psi \text{ or } i = j \\ 0 & \text{otherwise} \end{cases}$$

Note that LIM is a symmetric matrix. The conflict graph Ψ and the corresponding LIM for the network given in Fig. 1 is shown in Fig. 2.

As explained earlier, the conflict graph given in Fig. 2(a) comprises of three nodes, $N_{CG} = \{1, 2, 3\}$, corresponding to the communication sessions, $T = \{(s_i, d_i) \mid i \in \{1, 2, 3\}\}$, of the network given in Fig. 1. We emphasize here that the conflict graph represents the interference relations between different communication sessions and *does not* represent the network traffic flow. Throughout this paper, we will adopt this notion of *conflict graph* to characterize different network topologies based on interference between different communication sessions.

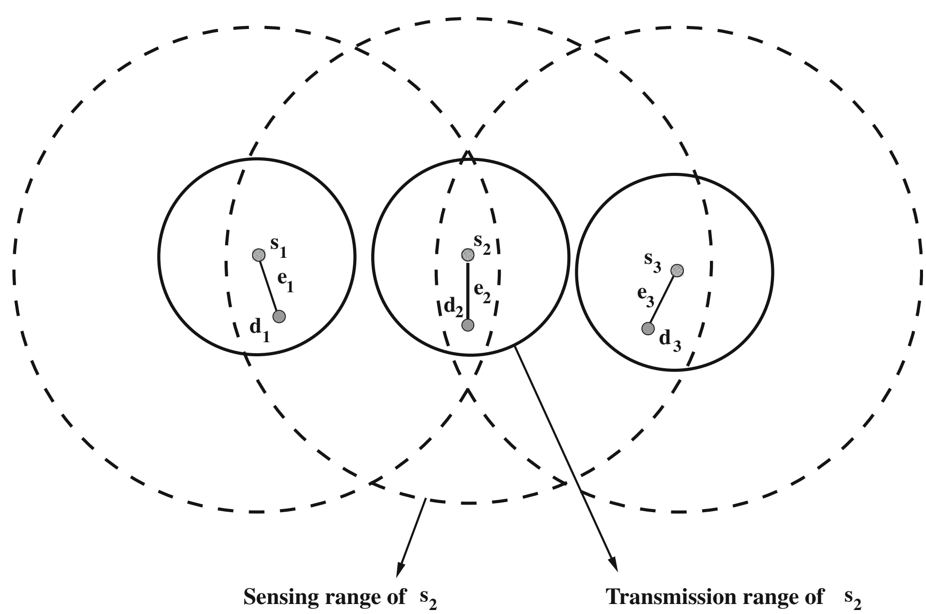
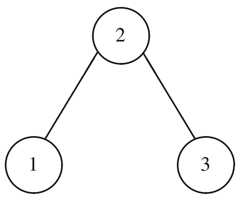


Fig. 1 A set of communication sessions in a multiple collision domain scenario



(a)

$$\begin{array}{c} \\ \mathbf{1} \\ \mathbf{2} \\ \mathbf{3} \end{array} \begin{array}{c} \begin{array}{ccc} \mathbf{1} & \mathbf{2} & \mathbf{3} \end{array} \\ \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix}_{3\times3} \end{array}$$

(b)

Fig. 2 (**a**) Conflict Graph for the network in Fig. 1. (**b**) The corresponding Link Interference Matrix (LIM)

3.4 A game-theoretic formulation of channel assignment problem

We now model the problem of channel assignment in an MCMR-MCD network (represented by the conflict graph, Ψ) as a non-cooperative game. Continuing further, due to the effect of interference between various nodes, the channel assignment for every node needs to be evaluated by taking its corresponding collision domain into consideration. In a non-cooperative game [22], each player is assumed to be rational in the sense that it is aware of all the alternatives, forms expectations about any unknowns, has clear preferences and chooses its action to maximize its utility. Players take into account their knowledge or expected behaviour of other players i.e., they reason strategically. In the current context, we consider each vertex $i \in N_{CG}$ in the conflict graph Ψ as a *rational player* whose aim is to maximize its data rate or throughput. Alternately, there is a one-to-one mapping from each rational player i in Ψ to a unique communication session $(s_i, d_i) \in T$ of the given network, Λ. Every player i in the non-cooperative game has a set of actions, A_i, from which it can make a choice. In the current context, an action $a_i \in A_i$ for a player i corresponds to a channel assignment scheme which gives all the channels on which it has been assigned radios. Let k_{ic} denote number of radios of player i that have been assigned a channel $c \in C$. We assume $k_{ic} \le 1$ as discussed before. Formally, we can define a_i which is the channel assignment for player i as follows.

$$a_i \stackrel{\text{def}}{=} (k_{i1}, k_{i2}, \ldots, k_{i|C|}) \tag{1}$$

We denote a channel assignment scheme for all N players as an *action profile*, $a = (a_1, \ldots, a_N) = (a_i)_{i \epsilon N_{CG}}$ which is a row vector in which column i of the vector corresponds to action of player i as given in Eq. 1. The set of all possible action profiles is represented by the set $A = \{(a_i)_{i \in N_{CG}} | a_i \in A_i, \forall i \in N_{CG}\}$. For any action profile, $a = (a_j)_{j \epsilon N_{CG}}$, and any $i \in N_{CG}$, we denote $a_{-i} = (a_j)_{j \in N_{CG} \setminus i}$ to be the list of actions (i.e., the channel assignments in this context) for all players except i. Conversely, given a list $a_{-i} = (a_j)_{j \in N_{CG} \setminus i}$ and an element a_i, we denote (a_i, a_{-i}) as the action profile $a = (a_i)_{i \epsilon N_{CG}}$.

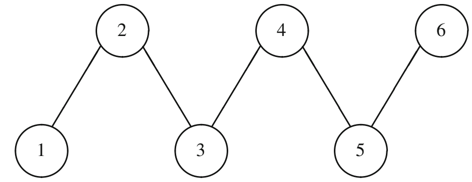


Fig. 3 An example network with the conflict graph representation

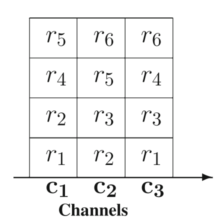
Before continuing further, we will illustrate the notations with the help of a simple example. Let us consider a conflict graph, Ψ, with vertex set $N_{CG} = \{1, 2, 3, 4, 5, 6\}$ as given in Fig. 3. Let us assume that each player $i \in N_{CG}$ has two radios ($k = 2$), each represented by r_i, and there are three channels for contention ($|C| = 3$). The LIM and an example channel assignment scheme for Ψ are given in Fig. 4. Based on our notations, we get $a_1 \equiv (k_{11}, k_{12}, k_{13}) = (1, 0, 1)$. This implies that player 1 has radios assigned to channels c_1 and c_3. Similarly, $a_2 \equiv (k_{21}, k_{22}, k_{23}) = (1, 1, 0)$; $a_3 \equiv (k_{31}, k_{32}, k_{33}) = (0, 1, 1)$ and so on. Collectively, the channel assignment scheme of Fig. 4(b) is given by the action profile $a = (a_1, a_2, a_3, a_4, a_5, a_6)$.

We denote the utility of every player $i \in N$ for a given channel assignment $a \in A$ by $U_i(a)$. For simplicity, we assume that the total data rate obtained by a player i from all channels that it has been assigned is an indicator of the player's utility. Every player will try to maximize its utility which results in maximizing the total data rate obtained from all the channels it has been assigned.

It is known that the maximum data rate possible in a channel $c \in C$ depends on the number of radios sharing the channel (say n_c). We denote this maximum data rate of channel c as $\tau^{max}(n_c)$. Let the data rate of a channel $c \in C$ when only one radio is using the channel be denoted by τ^c. Since all the channel characteristics are similar, we assume that $\tau^c = \tau^C$, $\forall c \in C$, where τ^C is a constant. It follows that $\tau^{max}(1) = \tau^C$.

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}_{6\times6}$$

(a)



(b)

Fig. 4 (**a**) The Link Interference Matrix (LIM). (**b**) A candidate channel assignment scheme, $a \in A$, for the network represented by the conflict graph in Fig. 3. Note that in this figure, $\{c_1, c_2, c_3\}$ represent the three channels under consideration and $\{r_1, r_2, ..., r_6\}$ represent the radios (two for every player) of the players $\{1, 2, ..., 6\}$. Each column corresponds to a particular channel and represent the radios that are assigned the corresponding channel. For example, in the figure, channel c_1 is assigned to radios r_1, r_2, r_4, r_5, channel c_2 is assigned to radios r_2, r_3, r_5, r_6 and so on

Now, in the current context, let K_{ic} be the number of radios, which have been assigned channel c, interfering with the radio of player i. For example, in the channel assignment scheme of Fig. 4(b), we can get $K_{11} = 2$, $K_{12} = 1$, $K_{13} = 1$. The maximum data rate obtained in channel c in the collision domain of player i can be denoted by $\tau^{max}(K_{ic})$. We assume that $\tau^{max}(K_{ic})$ is shared equally among the interfering radios which are assigned channel c. This fair rate assignment can be realized by using a MAC protocol like reservation-based time division multiple access (TDMA) schedule on a given channel. Fair sharing can also be attained using carrier sense multiple access/collision avoidance (CSMA/CA) protocol as shown by Cagalj et al. [11].

We divide the set of channels C into C_i and $\overline{C_i}$ where $C_i = \{c \mid k_{ic} = 1, \forall c \in C\}$ and $\overline{C_i} = C \backslash C_i$. Basically, C_i represents the set of channels where player i has been assigned radios. We know that, for a TDMA protocol and for CSMA/CA protocol which uses optimal back-off window values (as shown by Bianchi [23]), $\tau^{max}(K_{ic})$ is independent of K_{ic} and in this context, can be taken to be equal to τ^C. However, in practical CSMA/CA as implemented in the 802.11 standard [24], this assumption may not hold and thus, due to collisions, $\tau^{max}(K_{ic})$ will behave as a non-increasing function of K_{ic}. Thus, the utility, $U_i(a)$, for a player i when the action profile is a can be formulated as below.

$$
\begin{aligned}
&U_i(a) = \sum_{\forall c \in C_i} \tau(K_{ic}) \\
&\text{where, } K_{ic} = \sum_{\{j \mid LIM(i,j)=1\}} k_{j,c}, \quad \forall j \in N_{CG} \\
&\text{and, } \tau(K_{ic}) = \begin{cases} \left(\frac{1}{K_{i,c}} \times \tau^{max}(K_{ic})\right) & \text{if} \quad k_{ic} = 1 \\ 0 & \text{otherwise} \end{cases}
\end{aligned}
\tag{2}
$$

We explain the equations above using the example of Fig. 3. A summary of important notations is given in Table 1. Figure 4 gives the *LIM* and a channel assignment scheme, a, respectively, for the conflict graph (Ψ) of Fig. 3. For easier understanding of the example, let us assume $\tau^{max}(K_{ic}) = \tau^C$. The utility, $U_1(a)$, of player 1 in this example can be obtained as follows: $U_1(a) \equiv \tau(K_{1,1}) + \tau(K_{1,3}) = (1/2 + 1) \times \tau^C = 3/2 \times \tau^C$. Similarly, $U_2(a) \equiv \tau(K_{2,1}) + \tau(K_{2,2}) = \tau^C$; $U_3(a) \equiv \tau(K_{3,2}) + \tau(K_{3,3}) = \tau^C$.

Each player has a characteristic collision domain comprising of the nodes adjacent to it in the conflict graph Ψ. Hence we can obtain different *player-specific* channel assignment diagrams each from the perspective of an individual player. We illustrate two such diagrams for two

Table 1 Summary of important notations

Notation	Description
$\Lambda = (V, T)$	Input network with V vertices and T edges
i, j	Representation of indices from the set of natural numbers
$S = \{s_1, s_2, \ldots, s_m\}$	Source nodes in Λ
$D = \{d_1, d_2, \ldots, d_m\}$	Destination nodes in Λ
$e_i = (s_i, d_i)$	An edge (or communication session) in the network
k	Number of radios per node in Λ
$C = \{c_1, c_2, \ldots, c_p\}$	Set of channels in the network
$\Psi = (N_{CG}, E_{CG})$	Conflict graph with N_{CG} nodes and E_{CG} edges
N	The cardinality of the set N_{CG}
e_{ij}	An edge in conflict graph Ψ
LIM	The $N \times N$ Link Interference Matrix for the conflict graph (Ψ)
k_{ic} (or $k_{i,c}$)	Number of radios of player i assigned a channel $c \in C$
A_i	Set of actions of player $i \in N_{CG}$
a_{-i}	The list of actions for all players except i
r_i	Radio transceiver of player i
$\tau^{max}(n_c)$	Maximum data rate of channel c having n_c radios
τ^c	Data rate of a channel $c \in C$ when only one radio is using the channel
K_{ic} (or $K_{i,c}$)	Number of radios assigned channel c and interfering with radio of player i
C_i	Set of channels where player i has been assigned radios
$U_i(a)$	Utility for a player i when the action profile is a
N_i	The number of links in Λ that interfere with a link l in Λ where $l = (s_i, d_i) \in T$
R_i	Number of radios which are assigned channels in the player i's collision domain

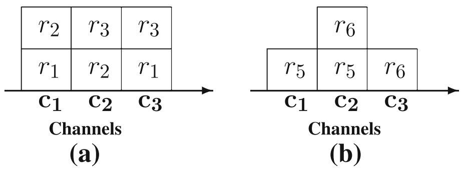


Fig. 5 Player-specific diagrams with respect to (**a**) player 2, (**b**) player 6, of channel assignment scheme given in Fig. 4

players (player 2 and player 6) in Fig. 5. Let N_i be the number of links in Λ that interfere with a link l in Λ where $l = (s_i, d_i) \in T$. Note that link l is associated with player i of the corresponding conflict graph Ψ. Thus, $N_i = (1 +$ degree of node i in the conflict graph $\Psi)$. We can see that $N_i = 2, \forall i \in \{1, 6\}$ and $N_i = 3, \forall i \in \{2, 3, 4, 5\}$ for the conflict graph of Fig. 3.

In the player-specific channel assignment diagrams, for every player $i \in N_{CG}$, as discussed earlier, we divide the set of channels C into $C_i = \{c \mid k_{ic} = 1, \forall c \in C\}$. It follows that $\overline{C_i} = C \backslash C_i$. Since there are N_i interfering players, we know that there are $R_i = (N_i \times k)$ radios which have to be assigned channels in the collision domain of player i. We now divide the number of competing radios R_i into R_i^+ and R_i^-, where

$$R_i^+ = \sum_{c \in C_i} K_{ic}; \quad R_i^- = \sum_{c \in \overline{C_i}} K_{ic} = (R_i - R_i^+) \tag{3}$$

Clearly, for the example in Fig. 5(a), $C_2 \equiv \{c_1, c_2\}$; $\overline{C_2} \equiv \{c_3\}; R_2^+ \equiv K_{2,c_1} + K_{2,c_2} = 4; R_2^- \equiv K_{2,c_3} = 2$.

4 Nash equilibrium and its properties in the channel assignment game

4.1 Concept of Nash equilibrium

In this section, we will introduce the fundamental concept of Nash equilibrium [22] and study its properties for the MCMR-MCD channel assignment game. We model the MCMR-MCD channel assignment with a single stage game represented by $\langle N, (A_i), (U_i)\rangle$ which corresponds to a fixed channel assignment among the players.

Definition 1 (*Nash equilibrium*) An action profile $a^* \in A$ is said to be an Nash equilibrium (NE) of a non-cooperative game, $\langle N, (A_i), (U_i)\rangle$, if for every player $i \in N$, we have $U_i(a_i^*, a_{-i}^*) \geq U_i(a_i, a_{-i}^*)$ for every action $a_i \in A_i$.

This implies that, if a^* is an NE, then no player i has an action yielding an outcome that it prefers to that obtained when it chose a_i^*, given that every other player j chooses its equilibrium action a_j^*. Hence no player i can *profitably deviate* from a_i^*, given the actions of the other players.

We will now establish necessary and sufficient conditions for NE and analyze the efficiency of the NE in this non-cooperative channel assignment game.

4.2 NE in MCMR-MCD channel assignment game

Lemma 1 *If a^* is an NE of the MCD channel assignment game, then $k_i = \sum_{c \in C} k_{ic} = k, \forall i \in N$.*

This implies that any player i in an NE will have to completely assign all its radios in order to achieve the maximum possible utility. In other words, each radio of the player i contributes positively to the overall utility of the player.

Now, consider any two arbitrary channels $c, d \in C$. We define a parameter $\delta_{i,c,d}$ for a player i as the difference between the number of interfering radios in channel c and channel d, i.e., $\delta_{i,c,d} = (K_{ic} - K_{id}), \forall c, d \in C$.

Theorem 1 *a^* is an NE for the MCMR channel assignment game, if and only if, for a player $i \in 1, 2, \ldots, N$,*

- *Case 1: $(N_i \times k) > |C|$ and $(k_{ic} \times \delta_{i,c,d}) \leq 1, \forall c, d \in C$ such that $k_{id} = 0$*
  OR
- *Case 2: $(N_i \times k) \leq |C|$ and $(k_{ic} \times K_{ic}) \leq 1\ \forall c \in C$*

Discussion In a single collision domain scenario (as considered in [17]), it is possible to give NE conditions that apply over the global channel assignment scheme [such as given in Fig. 4(b)], as in this case, every player in every channel is equally affected by congestion due to other players who are assigned the same channel. However, in a multiple collision domain setup, though there may be many players assigned to a particular channel (say $c \in C$), the effect of congestion on a single player (say player i) in c will be dependent on a subset of those players who are assigned c and need not be dependent on all the players who are assigned c. Only those players who are in the sensing range of player i will have an impact on the data rate of player i. Hence, we need a way to characterize this collision-domain specific effect of congestion in the evaluation of NE. The above theorem provides results for NE for a particular collision domain of player i. As we are considering a multiple collision domain scenario, the significance of the above theorem lies in the fact that it prescribes conditions for NE in every player's collision domain. Thus, the global channel assignment scheme (involving all the players) can be considered to be in NE if every collision domain specific channel assignment scheme (corresponding to a single player), such as given in Fig. 5(a) and (b), obeys the conditions evaluated by the above theorem. We now discuss the proof of the above result.

Proof

($\Rightarrow$) Case 1: Suppose $k_{ic} = 1$. We need to check if $\delta_{i,c,d} \leq 1$ is a necessary condition for NE for player $i \in 1, 2, \ldots, N$. We prove this case by contradiction. Let us assume that $\delta_{i,c,d} \geq 2$ for some channel $c, d \in C$ such that $k_{ic} = 1, k_{id} = 0$ and this represents an NE condition for the channel assignment game. Let a_i^* be the action of player i in this NE and the utility obtained by player i be denoted by $U_i(a_i^*, a_{-i}^*)$. Now if player i changes his action to a_i' in which he moves his radio from channel c to d, we can write the utility difference (Δ) obtained by player i as follows

$$\begin{aligned}\Delta &= U_i(a_i', a_{-i}^*) - U_i(a_i^*, a_{-i}^*) \\ &= (k_{ic} - 1) \times \tau(K_{ic} - 1) + (k_{id} + 1) \times \tau(K_{id} + 1) \\ &\quad - (k_{ic} \times \tau(K_{ic})) - (k_{id} \times \tau(K_{id})) \\ &= \tau(K_{id} + 1) - \tau(K_{ic}), \qquad \text{as} \quad k_{ic} = 1, k_{id} = 0\end{aligned}$$

Thus $\Delta > 0$ as $(K_{id} + 1) < K_{ic}$ due to our premise that $\delta_{i,c,d} \geq 2$. This is a contradiction to the assumption that a^* is an NE.

($\Leftarrow$) Case 1: Consider a channel assignment a that satisfies conditions of case 1 in the above theorem. We need to show that a is an NE. Now assume a player $i \in 1, 2, \ldots, N$ moves his radio from channel c to d. If $\delta_{i,c,d} = 1$ before changing the channel, then there is no change in utility after changing the channel because $K_{ic}^{old} = K_{id}^{new}$. If $\delta_{i,c,d} < 1$, then player i will get a lower utility after moving the radio as $K_{ic}^{old} < K_{id}^{new}$. Thus a player i cannot unilaterally change his strategy and increase his utility which shows that a is an NE. Hence the condition of Theorem 1 is also sufficient condition for NE in channel assignment game.

($\Rightarrow$) Case 2: Similar to the previous proof, we prove this condition by contradiction. Assume that $\exists c \in C$ such that $k_{ic} = 1$. Suppose in an NE, $(k_{ic} \times K_{ic}) > 1$ holds. Then, since the number of interfering radios is at the most equal to the number of channels, we can always find a channel c' to move radio of player i where $K_{ic'} = 0$. But the utility to player i in the latter case is more than the utility obtained in NE, which is a contradiction. Hence the condition of Theorem 1 is a necessary condition for NE in channel assignment game.

($\Leftarrow$) Case 2: Similarly, consider a channel assignment a which satisfies the condition $(k_{ic} \times K_{ic}) \leq 1$ $\forall c \in C$. Now, if $c \in C_i$, then moving radio of player i in channel c to another channel c' will not increase the utility of player i. Hence, player i cannot take any other action unilaterally and get a better utility. Hence a is an NE. Hence the condition of *Theorem* 1 is also sufficient condition for NE in channel assignment game. □

We can see that the above conditions hold true for all the collision domains in the example given in Fig. 4(b). Hence we can say that the global channel assignment scheme given in Fig. 4(b) is an NE. NE is an outcome of a game where players tend to be selfish in maximizing their utility. In the next section, we characterize the efficiency of an NE.

4.3 Efficiency of Nash equilibrium

Efficiency of an NE can be characterized through the concept of Price of Anarchy (PoA) as introduced by Koutsoupias and Papadimitriou [25], which compares the worst case NE to the socially optimal channel assignment. We establish a result below characterizing the lower bound that can be achieved for the PoA by considering the collision domain of player i. The PoA of a game is the ratio between the sum of the utilities of all players in a globally optimal solution compared to the sum of the utilities achieved in a worst-case NE (or alternatively, how much more cost does the NE bear with respect to the globally optimal). As we are considering benefits of players (i.e., sum of utilities) instead of costs throughout our work, it will be more appropriate to calculate the bounds on the inverse of PoA as PoA is usually evaluated with respect to costs incurred by the players. Hence, we will derive a lower bound on the inverse of PoA (denoted by κ) in our analysis i.e., $\kappa = 1/\text{PoA}$.

Theorem 2 *For a player $i \in 1, 2, \ldots, N$ and $c \in C$, if $((N_i \times k) > |C|)$ and $\tau^{max}(K_{ic})$ is a non-increasing function of K_{ic}, the (inverse of) price of anarchy (PoA) for the MCMR-MCD channel assignment game is at least $\left(\frac{\tau(\eta)}{\tau^{max}(1)}\right)$ where $\eta = \left\lceil \frac{(N \times k)}{|C|} \right\rceil$.*

Proof We have assumed that, in a channel, the maximum possible data rate is a non-increasing function of the number of radios allocated to that channel i.e., if more radios are allocated the same channel, the maximum possible data rate reduces from the channel perspective. Hence, in order to maximize network utilization, we should have a scenario wherein in every channel, the maximum data rate possible is achieved. This can be possible if there is only one radio allocated in every channel. Hence, the system optimal assignment is the state wherein, in every channel, there is only one radio allocated. Note that here we may not be able to assign channels to all radios of all players. Our objective is simply to maximize the utilization of the network regardless of whether every player has been satisfied with respect to his channel allocations or not.

So, if $\tau^{max}(K_{ic})$ is a non-increasing function of K_{ic}, then the maximum possible data rate will be achieved when K_{ic} is 1, $\forall c \in C$. In the MCMR-MCD channel assignment game, the social optimum is a situation where there is maximum utilization of all the channels from the network point of view i.e., the maximum data rate achieved from all channels $c \in C$ in the collision domain of player i is $\tau^{max}(1) \times |C|$.

We will now understand the worst case NE condition. In the worst case NE for a player i, the channel assignment will be such that there will be maximum interference possible in all the channels where player i has radios assigned, so that the net utility for player i will be the lowest. This can be achieved when the $N_i \times k$ radios are uniformly distributed among all channels $c \in C$. Formally, by using basic number-theoretic principles, for $\beta < |C|$, we can write

$$(N_i \times k) = ((\alpha \times |C|) + \beta) \Rightarrow \alpha = \left\lfloor \frac{N_i \times k}{|C|} \right\rfloor$$

It is important to note that this worst case NE should still conform to the conditions put forth by Theorem 1. Now, based on the parameter β, we have three possible scenarios.

- $\beta \geq k \Rightarrow R_i^+ = (\alpha + 1) \times k$
- $\beta < k \Rightarrow R_i^+ = (\alpha \times k) + \beta$
- $\beta = 0 \Rightarrow R_i^+ = (\alpha \times k)$

where R_i^+ is evaluated based on Eq. 3. We claim that, for any of the cases given above,

$$R_i^+ \leq \eta_i \times k \text{where } \eta_i = \left\lceil \frac{(N_i \times k)}{|C|} \right\rceil \tag{4}$$

Using these observations, we now proceed to derive the lower bound for κ for the MCMR-MCD channel assignment game.

$$\kappa \geq \frac{\text{Cumulative utilities in Worst Case NE}(A^*)}{\text{Cumulative utilities in system optimal assignment}}$$

In a system optimal assignment, each channel will have just a single radio assigned so that the maximum throughput of the channel is achieved (due to our assumption that maximum throughput is a non-increasing function of number of interfering radios). For the lower bound of inverse of PoA, we need to consider the case where the denominator is maximized. This is possible in a system optimal assignment where, in the collision domain of a player i, all its k radios will be assigned a separate channel.

$$\kappa \geq \frac{\sum_{i \in N_{CG}} U_i(A^*)}{\sum_{i \in N_{CG}} \tau^{max}(1) \times k} \geq \frac{\sum_{i \in N_{CG}} k \times \tau(\eta_i)}{\tau^{max}(1) \times k \times N}$$

Firstly, $\tau(\eta_i)$ is a non-increasing function and hence the numerator will be a minimum value because η_i is the maximum number of interfering radios in any channel $c \in C$ in a worst case NE (as given by Eq. 4). Secondly, in a worst case NE, there are equal number of interfering radios (η_i) in all channels where radios of player i have been assigned. Hence, for the lower bound of PoA, the numerator is minimized.

Let $\eta = \left\lceil \frac{(N \times k)}{|C|} \right\rceil$. We know that $\eta_i \leq \eta$ as $N_i \leq N$. We can write

$$\kappa \geq \frac{\sum_{i \in N_{CG}} k \times \tau(\eta)}{\tau^{max}(1) \times k \times N} = \frac{\tau(\eta)}{\tau^{max}(1)}$$

□

The above inequality is now independent of any collision domain specific parameters and thus, can be evaluated using the basic network parameters.

5 Fairness in MCMR-MCD channel assignment

In MCMR-MCD CA game, there may be multiple Nash Equilibrium (NE) outcomes and hence, it is necessary to characterize the quality of these NE outcomes. One important aspect of QoS in wireless networks is the notion of fairness. We will now assess the quality of NE outcomes by evaluating fairness of different NE allocations.

In a single collision domain scenario, all players are in the collision domain of each other and hence, it is reasonable to expect that the notion of fairness in such a setup corresponds to the utilities of all players being equal (assuming, of course, that all players have equal radios to be allocated). But in a multiple collision domain setup, every player has its own unique collision domain and hence, the notion of fairness is not straightforward to define in this setup as the throughput or utility obtained by the nodes need not be equal to one another and will depend on their corresponding collision domains.

Hence, we will define a measure of fairness called the *MCMR-MCD-fairness* and use this criterion to distinguish among different NE outcomes. In other words, we will classify different NE outcomes of the MCMR-MCD CA game either as *MCMR-MCD-fair* or not. We also derive necessary and sufficient conditions for an NE channel assignment to be termed *MCMR-MCD-fair* with respect to all the players in the MCMR-MCD CA game.

We now introduce some basic notation before proceeding with the fairness analysis. Let a^* be an NE channel assignment for the MCMR-MCD game. As explained in previous sections, we can easily generate the various player-specific channel assignment diagrams corresponding to this NE channel assignment a^*. For example, we know that Fig. 5 gives the player-specific channel assignment diagrams for player 2 and player 6 corresponding to the channel assignment given in Fig. 4 for the conflict graph given by Fig. 3.

Given any channel assignment scheme a (not necessarily an NE), let K^j_{ix} be the number of radios interfering with radio of player i in the player j-specific channel assignment diagram corresponding to channel $x \in C$. For example, from Fig. 5(a), we can get $K^2_{1c_1} = 2; K^2_{1c_2} = 1; K^2_{1c_3} = 1$ and so on. Let the set of all players in the collision domain of player i be represented by CD_i i.e.,

$$CD_i = \{j \in N \setminus \{i\} | LIM(i,j) = 1\}$$

Lemma 2 *In any NE channel assignment, a^*, of the MCMR-MCD CA game, $K^i_{ix} \geq K^j_{ix}, \forall i,j \in N, \forall x \in C$.*

Proof If not, then there is some $i \in N$ and $x \in C$ such that $K^i_{ix} < K^j_{ix} \implies$ There is some radio which interferes with radio i in collision domain of player j but not in collision domain of player i which is absurd. Hence the result. □

Now, given an NE channel assignment configuration, we will characterize the maximum utility that a player can obtain.

Definition 2 *Given an NE channel assignment, a^*, for the MCMR-MCD CA game, the best possible utility of a player $i \in N$ is given by*

$$u_i^{\max}(a^*) = \min_{j \in CD_i} u_i^j(a^*)$$

$u_i^j(a^*)$ is the utility of player i in the collision domain of player j. As explained in the MCMR-MCD utility model description, we have

$$u_i^j(a^*) = \sum_{x \in C_i} \frac{1}{K^j_{ix}} \tau(K^j_{ix}), \forall i \in N, \forall j \in CD_i$$

where $\tau(K^j_{ix})$ is a non-increasing function of K^j_{ix}.

It should be observed that $u_i^j(a^*)$ can be evaluated from the player-j-specific channel assignment diagram which, in turn, can be evaluated from the global channel assignment scheme as explained before.

The intuition behind the above observation is that a player in the multiple collision domain scenario is constrained in its payoff by its collision domain. If a player does not have any players in its collision domain, then it can use the whole bandwidth for itself. But, in a MCD scenario, a player will have possibly multiple interfering neighbouring players. As a result of that, a player will be constrained in its payoff by each of the collision domains of its interfering neighbours.

The distinction between $u_i^{\max}(a^*)$ and $u_i(a^*)$ should be made clear here. The former is a measure of the maximum utility that is *possible* for a player to achieve in a given NE configuration while the latter is the actual utility that the player is getting under the given NE configuration. Basically, if both of these utilities are the same for any player in the MCMR-MCD game under a given NE channel assignment, then we term the given NE configuration to be *MCMR-MCD-fair*. It is important to note here that there may exist *MCMR-MCD-unfair* NE configurations which do not satisfy the above property for all players and it will be our objective to understand these configurations in more detail in the rest of the section.

For notational ease, we will henceforth denote $u_i(a^*)$ by u_i, $u_i^{\max}(a^*)$ by $u_i^{\max}$ and $u_i^j(a^*)$ by u_i^j. It is understood that there is a NE channel assignment a^* behind the evaluations of these expressions. It is also understood that all NE channel assignment schemes satisfy the conditions specified by Theorem 1 (necessary and sufficient conditions for NE in MCMR-MCD CA game) as explained earlier.

For illustration purposes, consider the network given by Fig. 3. Here, player 5 has player 4 and player 6 in its collision domain. Now let us consider the collision domain of player 4 and player 6 separately (given in Fig. 6(a), (b), respectively). In the corresponding NE channel assignment given in Fig. 4(b), we can see that player 5 has been allocated channel c_1 and c_2. Assuming $\tau(k_x) =$ constant and normalized i.e., $\tau(1) = 1$, and observing channels c_1 and c_2 in Fig. 6, we get $u_5^4 = 0.5 + 1 = 1.5$ (as r_3 does not interfere with r_5) and $u_5^6 = 1 + 0.5 = 1.5$. The above observation states that the maximum utility of player 5 cannot be more than $min(u_5^4, u_5^6) = 1.5 = u_5^{max}$.

Now we will use the following notion of fairness for the MCMR-MCD channel assignment game.

Definition 3 An NE configuration a^* of the MCMR-MCD CA game is MCMR-MCD-fair if $u_i = u_i^{max}, \forall i \in N$.

Motivation There is a subtle observation to be made here. A player getting u_i^{max} in a NE channel configuration does not mean that this is the maximum payoff that the player in *any* NE channel configuration. It may happen that there are other NE configurations wherein the player may get better payoff. We are interested only in noting whether, given an NE channel assignment, *do all the players achieve their maximum utilities given their collision domain characteristics in the given NE configuration?* It turns out that it cannot be guaranteed that in any NE channel assignment

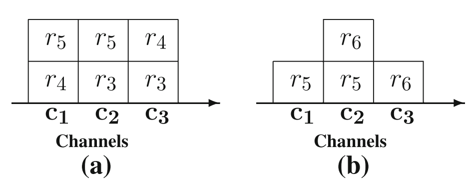


Fig. 6 Player-specific diagrams with respect to (**a**) player 4, (**b**) player 6, of channel assignment scheme given in Fig. 4

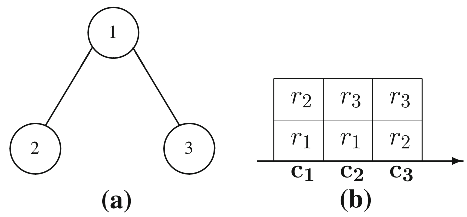


Fig. 7 (**a**) Example Conflict Graph. (**b**) An NE channel configuration

configuration of the MCMR-MCD game, all players achieve the maximum utility that is allowed by their corresponding collision domains. In other words, given an NE a^* it is not necessary that $u_i(a^*) = u_i^{max}(a^*), \forall i \in N$. To illustrate this, let us consider the following example

Figure 7(b) gives an NE channel assignment configuration for all the players for the corresponding conflict graph of Fig. 7(a). Assuming $\tau(k_x) =$ constant and normalized i.e., $\tau(1) = 1$, it can be seen that $u_1^2 = 1.5, u_1^3 = 1.5$. But we know that $u_1 = 1 \neq min(u_1^2, u_1^3) \Longrightarrow u_1 \neq u_1^{\max}$. Hence, such an NE configuration is not *MCMR-MCD-fair* to player 1 as it is not enabling player 1 to achieve the maximum utility that is possible in the NE configuration.

It can also be seen that if the number of available channels is increased to 4, then we can get an NE channel configuration in which each player achieves its maximum payoff possible in the configuration. The scenario is given in Fig. 8 where $u_1 = u_1^2 = u_1^3 = u_1^{max}$. It can be easily seen that similar conditions hold for player 2 and player 3. Hence, we see that this NE configuration is, in a multiple collision domain context, *fair* to all players in the game and thus, it is termed *MCMR-MCD-fair*.

We now derive necessary and sufficient conditions for an NE configuration to be *MCMR-MCD-fair*. As before, let K_{ix}^j denote the number of interfering radios of player i in the collision domain of player j associated with channel $x \in C$. We note that the below result is true for any non-increasing throughput function $\tau(k_x)$ where k_x is the number of channels allocated to channel $x \in C$.

Theorem 3 *Let a^* be an NE for the MCMR channel assignment game where a player can have at most one radio allocated to a channel. For any player $i \in N$, let C_i be the channels in which player i has been allocated a radio under a^*. For any player $i \in N$, the following condition is necessary and sufficient for a^* to be MCMR-MCD-fair.*

- $\sum_{x \in C_i} K_{ix}^i \geq \sum_{x \in C_i} K_{ix}^j$ *with equality holding for at least player $j \neq i$ where $j \in CD_i$.*

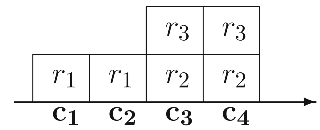


Fig. 8 NE configuration which is fair to all players

Proof ($\Leftarrow$) For any player $i \in N, \forall j \in CD_i$, suppose it is given that

$$\sum_{x \in C_i} K_{ix}^i \geq \sum_{x \in C_i} K_{ix}^j \tag{5}$$

We also know from Lemma 2 that, for a player $i \in N$,

$$K_{ix}^i \geq K_{ix}^j, \forall j \in CD_i, \forall x \in C_i \tag{6}$$

By hypothesis, let equality hold in Eq. 5 for at least one player $j \neq i$ where $j \in CD_i$. Using Lemma 2, we have

$$\begin{aligned} K_{iy}^i &= K_{iy}^j, \forall y \in C_i \Longrightarrow u_i = u_i^i = \sum_{y \in C_i} \frac{1}{K_{iy}^i} \tau(K_{iy}^i) \\ &= \sum_{y \in C_i} \frac{1}{K_{iy}^j} \tau(K_{iy}^j) = u_i^j \end{aligned} \tag{7}$$

Now, for other players $k \in CD_i$ where strict inequality holds in Eq. 5, we have by virtue of Lemma 2,

$$K_{iy}^i \geq K_{iy}^k, \forall y \in C_i \tag{8}$$

with strict inequality in Eq. 8 holding for at least one channel $z \in C$. This implies that

$$u_i = u_i^i < u_i^k \tag{9}$$

Combining Eqs. 7 and 9, we have

$$u_i = \min_{l \in CD_i} u_i^l = u_i^j = u_i^{max} \tag{10}$$

As choice of player i was arbitrary, we have that a^* is *MCMR-MCD-fair* by Definition 3.

($\Rightarrow$) Assume a^* is *MCMR-MCD-fair*. Now this implies that $u_i = u_i^{max} = min_{j \in CD_i} u_i^j, \forall i \in N$. This implies that there is at least one $j \in CD_i$ such that $u_i = u_i^j$. From Lemma 2, we have

$$\forall y \in C_i, K_{iy}^i \geq K_{iy}^j \Longrightarrow K_{iy}^i = K_{iy}^j \Longrightarrow \sum_{x \in C_i} K_{ix}^i = \sum_{x \in C_i} K_{ix}^j. \tag{11}$$

For all other players, $k \in N$, where $u_k > u_i^{max} = u_i$, it will be that $\exists y \in C_i, K_{iy}^i > K_{iy}^k$. Hence,

$$\sum_{x \in C_i} K_{ix}^i > \sum_{x \in C_i} K_{ix}^k \tag{12}$$

Combining Eqs. 11 and 12, we have

$$\sum_{x \in C_i} K_{ix}^i \geq \sum_{x \in C_i} K_{ix}^j, \forall i \in N, \forall j \in CD_i \tag{13}$$

□

One observation immediately follows from the above result. For a player $i \in N$, if, for every $j \in CD_i$, there exists at least one $y \in C_i$ such that $K^i_{iy} > K^j_{iy}$, then clearly utility value of player i in collision domain of player j is more than the corresponding utility value in its own collision domain with respect to channel y. Hence, player i will not achieve the maximum allowed utility in the given NE channel assignment. This implies $\sum_{x \in C_i} K^i_{ix} > \sum_{x \in C_i} K^j_{ix}, \forall j \in CD_i$. In this case, we can say that the NE configuration is not *MCMR-MCD-fair*.

6 Convergence to Nash equilibrium

We now propose four channel assignment algorithms, each in a different setting, for a given MCMR-MCD network. We consider both centralized and distributed decision making scenarios. Further, the algorithms differ based on how *informed* each player is about the number of interfering radios assigned in all channels in the current assignment. The first of these algorithms is a centralized algorithm which uses *perfect information* about interfering radios in the collision domains of all players in all channels and comes up with a channel assignment scheme in which the utility for the players in maximized. Next, we propose a distributed algorithm with perfect information in which the channel assignment scheme is decided in a distributed way by each of the nodes of the network, taking into consideration their corresponding collision domains. This implies that every node has information about all the interfering radios in *all* channels. Next, we considered a more restricted version of the distributed algorithm which has imperfect information about the interfering radios in the channels. This algorithm can use information only from the channels in which it has radios currently assigned and has to form some expectation or belief about the occupancy of other channels. Lastly, we consider using a no-regret learning-based algorithm in a uninformed setting in which players do not have any knowledge of the interfering radios in the channels. The players essentially try to arrive at better channel assignment schemes based on the rewards they get for using each of its strategies. We will now investigate each of these algorithms in more detail in the following sections.

6.1 Centralized algorithm using perfect information

The pseudo-code for the centralized algorithm is given in Algorithm 1. As illustrated, this algorithm assigns radios to each player in a sequential manner and hence, this algorithm needs a centralized control where the assignment is done by globally co-ordinating with all the players. It requires the complete conflict graph of the network representing the interference characteristics. Also, it assumes that, for every channel assignment of a player, perfect information about its interfering radios in all the channels is available. Given this information, the algorithm proceeds by assigning radios to the channels where minimum interference is caused. As the assignment scheme is sequential by nature, an advantage of this scheme is that the assignment of radios for a player is one-time and does not change due to other players' assignments. However, in practical scenarios, global coordination in an wireless network with selfish players cannot be expected. Secondly, the assumption about perfect information about all the interfering radios in all channels for every player may not be pragmatic. We now relax on the centralized nature of the algorithm and present a distributed version of the algorithm keeping the second assumption true.

Algorithm 1 Centralized NE Channel Assignment with Perfect information

Require: $N \neq \phi$, $k > 0$, $|C| > 0$, $k < |C|$
Ensure: The channel assignment is a Nash equilibrium
for all $i \in N$
  for $j \leftarrow 1$ to k
    Calculate the K_{il} values, $\forall l \in C$, for the current channel assignment
    if $K_{ic} == K_{il}$, $\forall c, l \in C$ **then**
      Update the channel assignment scheme by assigning radio j of player i on channel c where $k_{ic} = 0$
    else
      Update the channel assignment scheme by assigning radio j of player i on channel c where $c = \arg min_{l \in C} K_{il}$ and $k_{ic} = 0$
    end if
    Set the updated channel assignment scheme as the current channel assignment scheme.
  end for
end for

6.2 Distributed algorithm using perfect information

In this section, we present a distributed version of the channel assignment algorithm. In this setup, the players initially tune their radios to some random channels and from then on the algorithm proceeds in a *round-based* manner. As the assignment scheme is static by nature, we assume that every player is assumed to know the players in its collision domain. In every round, a node may get an opportunity to change its channel assignment based on a selection scheme which is similar to the IEEE 802.11 back-off mechanism which uses a back-off counter initialized with a random value chosen with uniform probability from the set $\{1, \ldots, W\}$. We note here that this round-based approach, which is applied with channel assignment algorithms in the literature like [17, 18], is used to ensure the distributed algorithm runs with some arbitrary ordering among the nodes. This is done to avoid unstable channel configurations resulting from channel assignment algorithms with no ordering as mentioned in [17]. An intuition behind these unstable channel assignments is that if there is no ordering among the nodes, then every node may consider a stale channel assignment scheme to make its current channel reconfigurations and this may lead to recurring channel reconfigurations which in turn will lead to the algorithm alternating infinitely among a few channel configurations.

Algorithm 2 Distributed NE Channel Assignment with Perfect Information

Require: $N \neq \phi$, $N_i > 0$, $k > 0$, $k < |C|$
Ensure: The channel assignment is a Nash equilibrium
Get a random channel assignment
while not in NE **do**
  Get the current channel assignment
  for all $i \in N$ **do**
    if backoff$_i$ == 0 **then**
      {/* Establish NE*/}
      if $(N_i \times k) > |C|$ **then**
        for $j \leftarrow 1$ to k **do**
          Assume that radio j uses channel b. Find c_{min} where $c_{min} \leftarrow arg\,min_{c \in C \setminus C_i} K_{ic}$
          if $(K_{ib} - K_{ic_{min}}) > 1$ **then**
            Move radio j from channel b to channel c_{min}
          end if
        end for
      else
        for $j \leftarrow 1$ to k
          Assume that radio j uses channel b
          if $(k_{ib} \times K_{ib}) > 1$ **then**
            Move radio j from channel b to a channel c uniformly chosen from $\overline{C_i}$
          end if
        end for
      end if
      backoff$_i$ $\leftarrow$ number sampled with uniform distribution from {1, 2, ..., W}
    else
      backoff$_i$ $\leftarrow$(backoff$_i$ $- 1$)
    end if
  end for
end while

Every round decrements the back-off counter by one and once a node is in a round where its back-off counter decrements to zero, it will evaluate perfect information about all the interfering radios currently tuned to *all* the channels in its collision domain. One of ways to get this information is by using an extra radio per player which scans all the channels and collects relevant information. Based on this information, it will use the result established in Theorem 1 and change its assignment to a better reorganization of its radios. The back-off counter is then reset to a random value and the process is repeated in a similar way eventually leading to a rationally beneficial NE. Due to this back-off mechanism, the players change their channel assignments almost in a sequential manner. The pseudo-code of the algorithm is presented in Algorithm 2. We further show later in our simulations that this algorithm converges to an NE.

As mentioned earlier, knowing the current channel assignment of interfering users in *all* channels maybe difficult and thus, we relax this assumption and present another distributed algorithm which is based on imperfect information about the channel assignment.

6.3 Distributed algorithm using imperfect information

This distributed algorithm is similar to the one presented in the previous section except that here, the players do not have all the information pertaining to channel assignment in all the channels. Each player evaluates the *imperfect* information of interfering radios *only* in the channels where its radios are currently tuned. Before presenting the algorithm, we will develop some analytical results based on the imperfect information of each player.

Proposition 1 *In the MCMR-MCD channel assignment game with imperfect information, the maximum possible number of interfering radios for a player i in any assigned channel in an NE is at most* $\left(\left\lfloor\frac{N_i \times k + |C| - k - R_i^+}{|C| - k}\right\rfloor\right)$.

Proof We will arrive at an upper bound on the maximum possible number of interfering radios for a player i in any assigned channel in an NE channel assignment scheme for the given MCD network. Consider a channel assignment for player i. Let $\alpha = \max_{a \epsilon C_i} K_{ia}$. Since number of radios in the interference neighbourhood of player i is $(N_i \times k)$ which is a constant, we know that the total number of radios, i.e., $(R_i^+ + R_i^-)$ (from Eq. 3), in the collision-domain specific channel assignment diagram is a constant. We can say that $R_i^- = ((N_i \times k) - R_i^+)$. Applying Theorem 1, we know that $min_{a \in \overline{C_i}} K_{ia} = max_{a \in C_i} K_{ia} - 1 = (\alpha - 1)$. Now,

$$\begin{aligned} R_i^- &\geq ((\alpha - 1) \times (|C| - k)) \\ (N_i \times k - R_i^+) &\geq ((\alpha - 1) \times (|C| - k)) \end{aligned} \tag{14}$$

$$\begin{aligned} \Rightarrow \alpha &\leq \left(\frac{N_i \times k + |C| - k - R_i^+}{|C| - k}\right) \\ &\leq \left(\left\lfloor\frac{N_i \times k + |C| - k - R_i^+}{|C| - k}\right\rfloor\right) \end{aligned} \tag{15}$$

As we have no knowledge of channels not occupied by player i (imperfect information), we convert R_i^- in terms of

R_i^+ as shown in Eq. 14. The simplification in Eq. 15 is due to the fact that α is an integer. Based on the imperfect information available to each player, we use the above bound on α for the evaluation of the current channel assignment and modify it to lead to an NE. We note here that, due to imperfect information available to each player, there may arise some inefficient channel assignment configurations which the player may misunderstand to be an NE. So, in order to come out of such unstable NE configurations, we introduce a perturbation factor, ε, which is a very small value (10^{-4} in our simulations) and represents the probability with which a player will change his channel assignment even if it satisfies conditions of Proposition 1. We present the pseudo-code of the algorithm in Algorithm 3. □

Algorithm 3 Distributed NE Channel Assignment with Imperfect Information

Require: $N > 0, N_i > 0, k > 0, k < |C|, R_i^+ > 0$
Ensure: The channel assignment is a Nash equilibrium
Get a random channel assignment
while not in NE **do**
  Get the current channel assignment
  for all $i \in N$ **do**
    if backoff$_i$ == 0 **then**
      {/* Establish NE*/}
      $\mu \leftarrow (((N_i \times k) + |C| - k - R_i^+)/(|C| - k))$
      if $\max_{d \epsilon C_i} K_{id} > \mu$ **then**
        for $j \leftarrow 1$ to k **do**
          Assume that radio j uses channel b
          if $K_{ib} > \mu$ **then**
            Move radio j from channel b to channel $c \in \overline{C_i}$, where c is chosen with uniform probability from $\overline{C_i}$.
          end if
        end for
      else
        for $j \leftarrow 1$ to k **do**
          Assume that radio j uses channel b
          if $K_{ib} > \mu$
            Move radio j from channel b to channel $c \in \overline{C_i}$ with probability ε, where c is chosen with uniform probability from $\overline{C_i}$
          end if
        end for
      end if
      backoff$_i$ ← number sampled with uniform distribution from {1, 2, ..., W}
    else
      backoff$_i$ ←(backoff$_i$ − 1)
    end if
  end for
end while

6.4 No-regret learning for MCMR-MCD channel assignment game

All the three algorithms proposed above can be termed as *informed* by nature in the sense that they depend on the knowledge of the other players in the game and their utilities. They also rely on explicit communication between interfering nodes while deciding on a channel assignment scheme in every round. Also, the distributed algorithms proceed in a sequential manner with each player getting an opportunity to change its channel assignment only when its turn arrives. Although such informed settings can lead to stable NE schemes in the MCMR-MCD channel assignment game, there may exist scenarios where such information may not be assumed. Here, we will explore the possibility using *learning* algorithms to arrive at stable channel assignment for the MCMR-MCD game in a *uninformed* setting where players are unaware of other players in the game and their strategies. To be specific, we will use a class of learning algorithms known as *no-regret* learning algorithms and study their behaviour in the channel assignment game. No-regret learning algorithms allow initially uninformed players to acquire knowledge about the state of their environment they are in as the game is played in a repeated way. These algorithms have the advantage that each player need not explicitly know the number of players in the game and the strategies used by them. Further, the players need not be aware of the utility functions of other players. They will learn to play better strategies based on the rewards obtained from playing each of its strategies. In this work, we will focus on an important randomized no-regret learning algorithm known as the Freund and Schapire Informed (FSI) algorithm [26] which is a variant of the Littlestone and Warmuth's [27] weighted majority algorithm. Due to space constraints, we explain the core aspects of the algorithm below. The reader is directed to [26] and [28] for more details.

One of the measures of the performance of a learning algorithm is the *regret*, which is the expected value of the difference between total utility of the algorithm and the total utility of the best action. The concept of regret involves the benefits a player feels after taking a particular action compared to its other possible actions. The actions which result in lower regret values will be updated with higher probabilities and hence, ultimately actions that are more rewarding will be used more often in the long run i.e., the algorithm *learns* to play profitable actions. It has been shown in [26] that FSI suffers a regret of at most $\sqrt{2Tln K}$ where T is the maximum number of rounds and K is the number of strategies available to a player.

The FSI algorithm works in a round-based manner. In a round t, each player i maintains a probability distribution,

p_i^t, on the set of its strategies, together denoted by the profile $p^t = (p_1^t, p_2^t, \ldots, p_N^t)$. The players choose an action profile denoted by $A^t = (a_1^t, a_2^t, \ldots, a_N^t)$ based on their corresponding probability distributions of actions i.e., in round t, player i chooses action a_i^t based on its probability distribution p_i^t. Based on the action profile, A^t, chosen by the players in round t, rewards (or utilities) are obtained for each of the actions of the players at the end of the round. Further, this learning algorithm is called *fully informed* in the sense that it is assumed that every player on each round has access to the utility associated with all of his strategies (not just the one that was chosen in that round).

We should however remark here that the term *information* in the context of no-regret learning algorithms corresponds to a player being aware of the utilities of all its possible strategies and not just the strategy it chose in the current round. This is different in the case of the previous distributed algorithms discussed in the paper where *information* signifies the players being aware of knowledge of other players and their actions.

The algorithm, also termed as *Hedge*, utilizes the cumulative utility obtained by player i over time t if he chooses action a_i given that other players had played a_{-i}^t, for every $a_i \in A_i$. We will denote the cumulative utility obtained by player i for choosing action a_i in all the t rounds as $U_i^t(a_i) = \sum_{j=1}^{t} U_i(a_i, a_{-i}^j)$. At the end of every round, the algorithm updates weights associated with each action based the cumulative utility function described above and re-calculates the probability distribution associated with the set of its strategies for the next round. Let $p_i^{t+1}(a_i)$ be the probability of choosing a_i action by player i in round $t+1$. The main idea of *Hedge* is to simply choose an action a_i at time t with probability proportional to $(1+\alpha)^{U_i^t(a_i)}$. Thus actions yielding high rewards quickly gain a high probability of being chosen. For some $\alpha > 0$, the probability distribution of actions of player i in round $t+1$, $\mathbf{p_i^{t+1}}$, is calculated as follows,

$$p_i^{t+1}(a_i) = \frac{(1+\alpha)^{U_i^t(a_i)}}{\sum_{a_i' \in A_i} (1+\alpha)^{U_i^t(a_i')}} \tag{16}$$

So, an important questions we investigate is—*Will informed no-regret learning algorithm like Hedge converge to stabilizing outcomes in the MCMR-MCD channel assignment game?* We later find out through experimental evaluation that, in the MCMR-MCD channel assignment game, *Hedge* does lead to a stabilizing strategies where players choose a probability distribution over possible actions. Hence, the no-regret learning approach can be applied in scenarios where players are unaware of other players' actions. We will now study the no-regret algorithm along with other algorithms proposed in the paper in more detail through extensive simulations.

7 Performance evaluation

7.1 Simulation environment

We first considered a *informed* setting where every player is aware of the existence of its interfering players and their strategies in the network. An MCMR network with a 10-node conflict graph representation given in Fig. 9 was chosen for simulations and performance of the three algorithms in the informed setting namely Algorithm 1, Algorithm 2 and Algorithm 3 was studied for this example network by implementing them in MATLAB. It is to be noted here that we have not shown the actual network setup used in simulations due to space limitations. As there is a *one-one* relationship with the underlying network and the corresponding conflict graph, it is assumed that there is a unique MCMR network consisting of 10 transmitting nodes which can be represented by the conflict graph given in Fig. 9. As per our assumptions, each transmitting node has a corresponding receiver node and hence, the underlying network is a network of ten transmitter-receiver pairs having the interference characteristics as shown in Fig. 9. We considered 8 orthogonal channels as default value for $|C|$ in our simulations.

The example network was chosen to depict the multiple collision domain aspect of the underlying network. In the conflict graph of Fig. 9, we can observe that there is diversity in the number of interfering neighbours of any node. For example, node 1 and node 2 have high interference from their neighbourhood (as much as 6 interfering nodes) while node 7 and node 8 have low interference (2 interfering nodes). As this work is related to channel assignment in such diverse interference neighbourhoods, the example conflict graph given in Fig. 9 can be used to

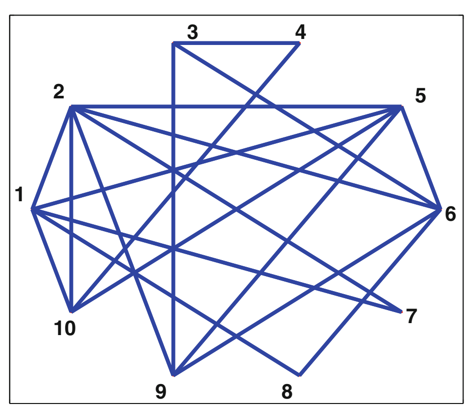


Fig. 9 Example 10-node conflict graph used in simulations

obtain good understanding of the performance of the proposed channel assignment algorithms.

Next, we characterized different 10-node networks and studied efficiency of these algorithms for these random network topologies. Finally, we considered a *uninformed* setting where every player is unaware of its interfering players and their strategies. In this setup, the no-regret algorithm, *Hedge*, was implemented (in MATLAB) and performance evaluation was done by executing *Hedge* in all the nodes of the example network of Fig. 9.

In the simulations, algorithms were executed for several runs to get different estimates for the measured metrics. In every run, every algorithm executed for a number of pre-configured rounds (denoted by parameter T) and a 95% confidence interval on the mean value of measured metrics was calculated after obtaining the estimates for all the runs and the results were plotted. More details on the execution of the algorithms are given in the following sections. Before we present our observations, we define metrics to evaluate the performance of the algorithms. We set $W = 15$ and $\varepsilon = 10^{-4}$ in all our simulations.

7.2 Performance metrics

Definition 4 (*Convergence Index*) Given a channel assignment $a \in A$, we define the convergence index, $\gamma(a)$, which gives an indication of how many players satisfy the conditions put forth by Theorem 1.

$$\gamma(a) = \sum_{i \in N} \sum_{(c,d) \in L_i} \mathbf{1}_{\delta_{i,c,d} \leq 1} \tag{17}$$

where

$$L_i = \{(c,d) | k_{ic} = 1, k_{id} = 0, \forall c, d \in C\}$$
$$0 \leq \gamma(a) \leq (k \times (|C| - k))$$

and $\mathbf{1}_{\delta\_i,c,d} \leq 1$ is the characteristic function defined as below.

$$\mathbf{1}_{\delta_{i,c,d} \leq 1} = \begin{cases} 1 & \text{if } \delta_{i,c,d} \leq 1 \\ 0 & \text{otherwise} \end{cases} \tag{18}$$

We consider only channel pairs, (c, d), such that $k_{ic} = 1$, $k_{id} = 0$ and use these pairs (represented by the set L_i) to check if they satisfy the condition put forth from Theorem 1 i.e., $(k_{ic} \times \delta_{i,c,d}) \leq 1$. Other channel pairs, where $k_{ic} = 0$, trivially satisfy this condition and hence, we do not consider them in the evaluation of $\gamma(a)$. We know that, for every player i, there are $(k \times (|C| - k)) = |L_i|$ channel pairs which are needed to evaluate Eq. 18. Basically, $\gamma(a)$ gives an indication of how close the channel assignment a is to the expected NE channel assignment.

Definition 5 (*MCD-Efficiency*) Given a channel assignment $a \in A$ and an NE channel assignment $a^* \in A$, the MCD-efficiency, ω, is defined as

$$\omega(a) = \frac{\gamma(a)}{\gamma(a^*)} \tag{19}$$

We note here that $0 \leq \omega(a) \leq 1$. We claim that the metric $\omega(a)$ is suited for evaluating the channel assignment obtained from the execution of proposed algorithms in Sect. 6 since it uses the result of Theorem 1 in determining the convergence properties.

Definition 6 (*Average MCD-efficiency and MCD-efficiency ratio*) The average MCD-efficiency, $\bar{\omega}$, at round T is defined as, $\bar{\omega}(a,T) = \frac{\sum_{t=1}^{T} \omega(t,a)}{T}$, where $\omega(t, a)$ is $\omega(a)$ in round t. The MCD-efficiency ratio (Ω) metric is defined as $\Omega = \lim\inf_{T\to\infty} \bar{\omega}(a,T)$.

7.3 Observations from simulations

Firstly, we will evaluate the performance of the three *informed* algorithms namely Algorithm 1, Algorithm 2 and Algorithm 3. Later, we will study the behaviour of the no-regret learning algorithm (*Hedge*). The results obtained from our simulations for the three informed algorithms are given in Figs. 10–22.

7.3.1 Efficiency analysis

Figures 10 and 11 plot the behaviour of ω with time for the distributed algorithms, Algorithm 2 and Algorithm 3. We can observe that ω *converges* to 1 after starting from a random channel assignment. However, due to imperfect information available, convergence to 1 is not reached for Algorithm 3. It can be observed that Algorithm 3, though it is more constrained due to imperfect information available to it, performs quite well when it starts with different random channel assignments. As these plots were taken after starting from random channel assignments, we measured the *MCD-efficiency ratio*, Ω, to get an idea over the performance of the algorithms. Simulations were performed by varying the number of radios per device (k). For each value of k, 100 different estimates of the metric were collected. Each estimate was obtained by starting from a random channel assignment. As mentioned before, a back-off counter per player (similar to the approach used in 802.11b protocol [1]) was used to provide sequential opportunities to *change* its corresponding channel assignment. One round is completed when all the nodes have been given one opportunity to decrement their back-off counters. During a round, if the back-off counter for any

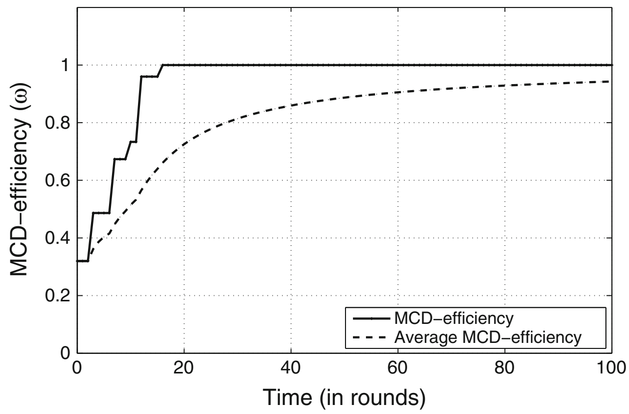


Fig. 10 Time versus MCD-efficiency for Algorithm 2 with $N = 10$, $k = 3$, $|C| = 8$

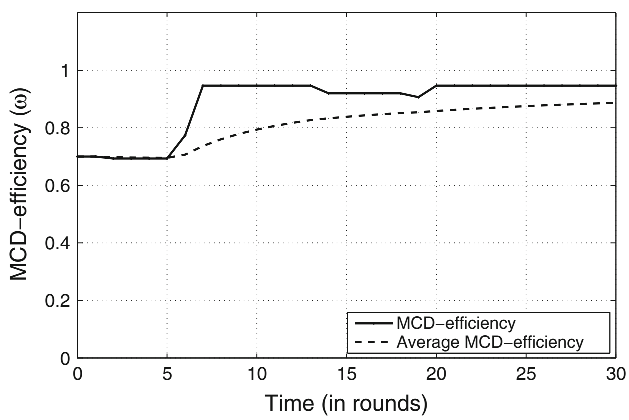


Fig. 11 Time versus MCD-efficiency for Algorithm 3 with $N = 10$, $k = 3$, $|C| = 8$

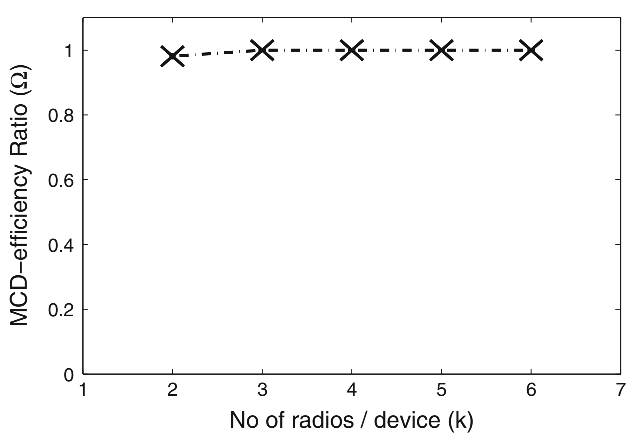


Fig. 12 No. of radios/device (k) versus MCD-efficiency ratio (Ω) for Algorithm 2 with $N = 10$ and $|C| = 8$

player reaches 0, then the updating algorithm for that player checks for NE conditions and updates the channel assignment if necessary. The player then resets the back-off counter. The algorithm is executed for $T = 10000$ rounds

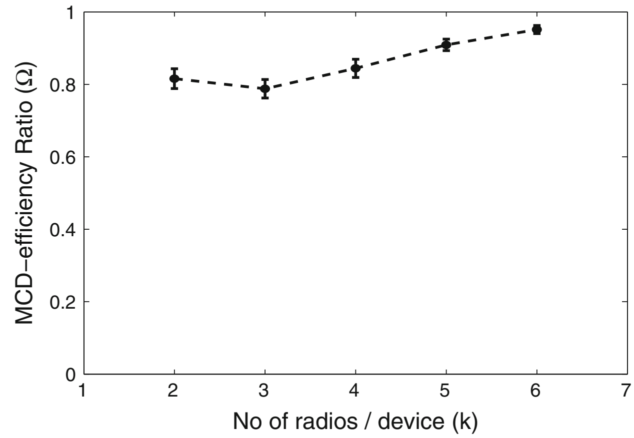


Fig. 13 No. of radios/device (k) versus MCD-efficiency ratio (Ω) for Algorithm 3 with $N = 10$ and $|C| = 8$

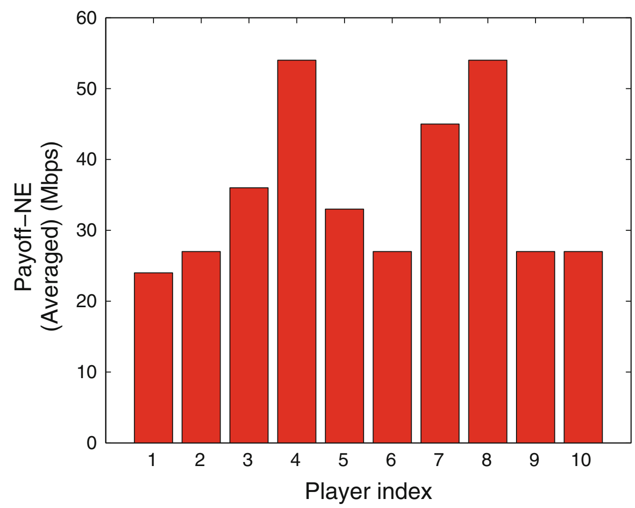


Fig. 14 PlayerIndex versus NE Payoff (from Algorithm 1) for $N = 10$, $k = 3$ and $|C| = 8$

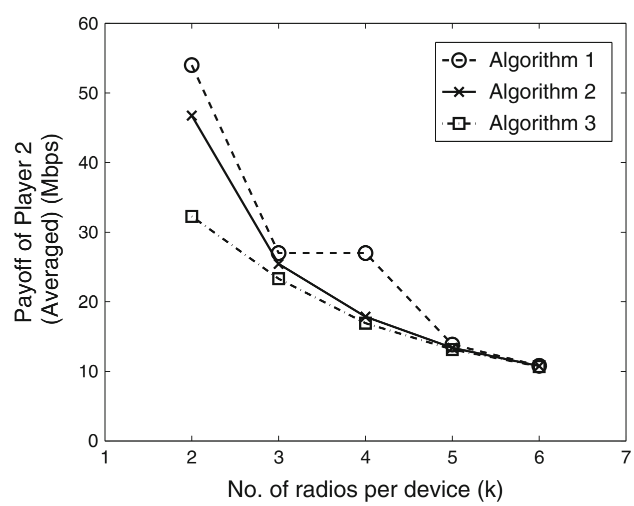


Fig. 15 No. of radios per device (k) versus Converged-NE Payoff for Player 2 with $|C| = 8$

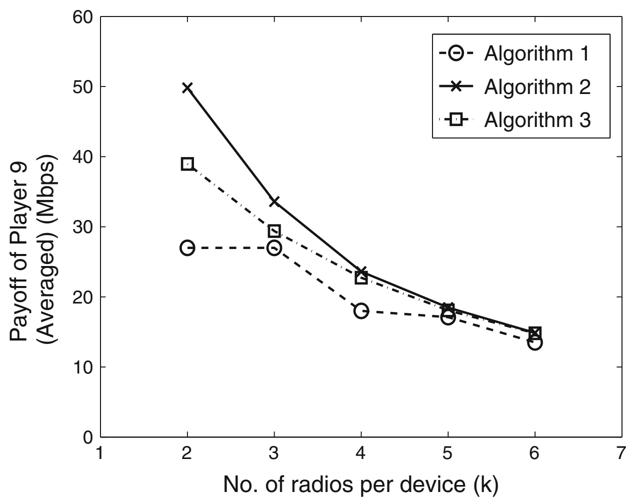


Fig. 16 No.of radios per device (k) versus Converged-NE Payoff for Player 9 with $|C| = 8$

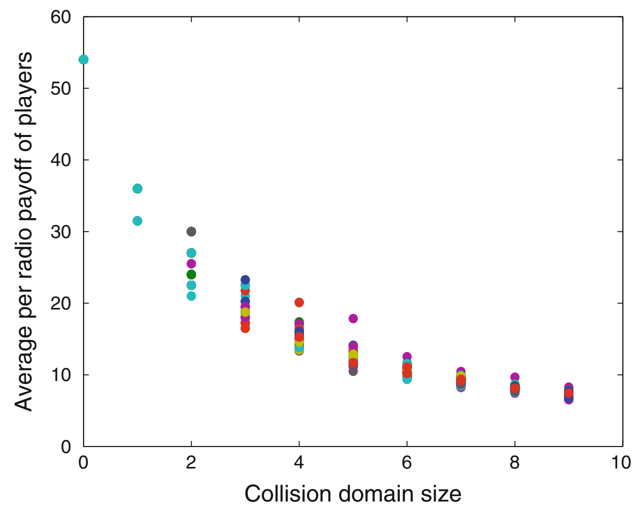


Fig. 18 Average Payoff of Players (Algorithm 3) versus Collision domain size with $N = 10$, $k = 6$ and $|C| = 8$

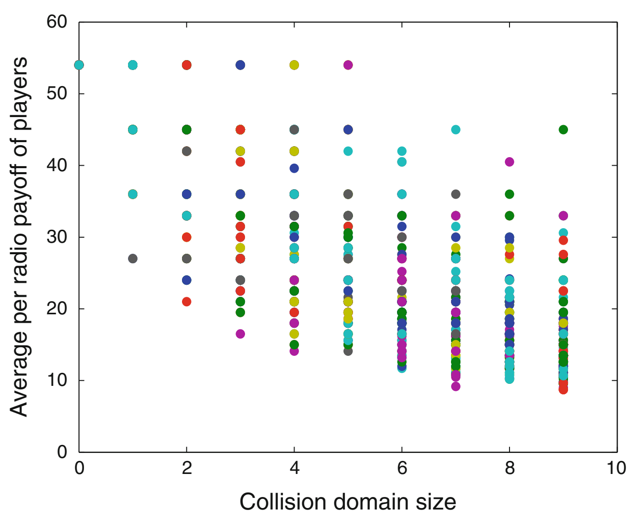


Fig. 17 Average Payoff of Players (Algorithm 3) versus Collision domain size with $N = 10$, $|C| = 8$ and $k = 3$

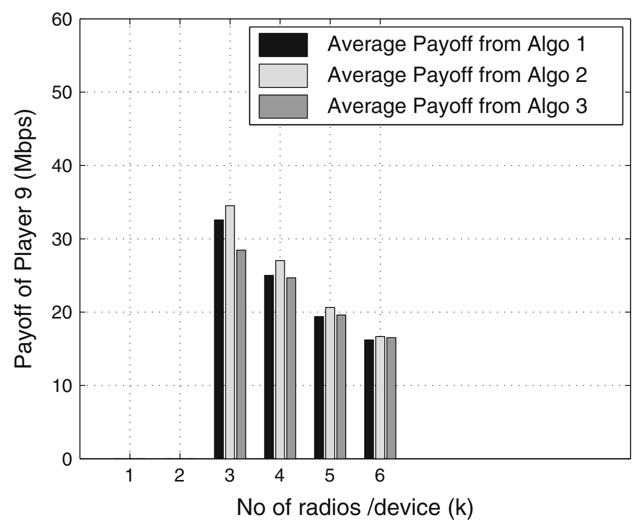


Fig. 19 Average Payoff of Player 9 versus number of radios per device with $N = 10$, $|C| = 8$ and $k = 3$

and Ω was calculated. This constituted one estimate of Ω. A 95% confidence interval on the mean value of Ω for 100 runs was calculated and the results were plotted. Similar experiments were run for different values of k. The results obtained for each of the algorithms are shown in Figs. 12 and 13.

We can observe that Ω is closer to 1 for Algorithm 2 than Algorithm 3 due to the full information available in Algorithm 2 in deciding the channel assignment, while Algorithm 3 uses a probabilistic approach to compensate for the imperfect information available to it. We should note here that both the algorithms try to converge to an NE by using the conditions established by Theorem 1.

It can also be observed that Ω tends to increase with k due to more information made available on the channels having interfering radios.

7.3.2 *Payoff analysis*

We will now analyze the utilities obtained by the players as a result of running the three algorithms. Experiments were conducted by varying the value of k between 2 and 6. For each value of k, utilities of the players were obtained for 100 runs of the experiment and then, the average of these utilities was plotted. We provide an example plot for $k = 3$ in Fig. 14. We further plot the utility values for different

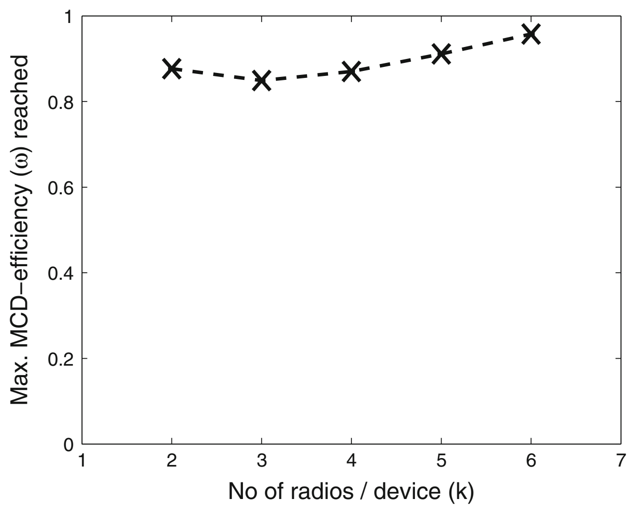


Fig. 20 No. of radios/device (k) versus Max. MCD-efficiency (ω) reached for $N = 10$ and $|C| = 8$

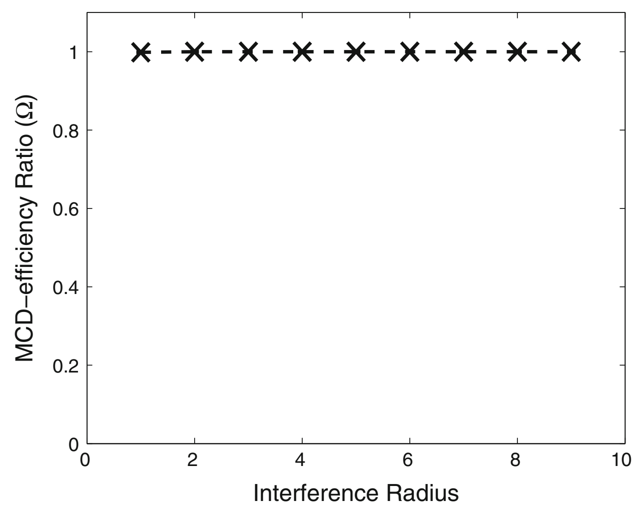


Fig. 22 Interference Radius (IR) versus MCD-efficiency ratio (Ω) for $N = 10$, $|C| = 8$ and $k = 3$

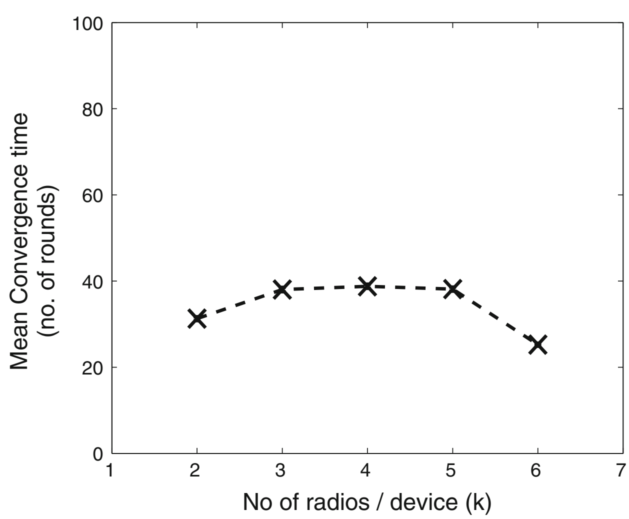


Fig. 21 Number of radios/device (k) versus mean convergence timefor $N = 10$, $k = 6$ and $|C| = 8$

values of k for the three algorithms with respect to Player 2 and Player 9 in Figs. 15 and 16, respectively.

First of all, we can observe from Fig. 14 that NE in a MCD scenario *does not* necessarily lead to equal utility distribution among the players as observed in [17] for the single collision domain scenario. This can be attributed to the variation in the number of interfering users for every node. In a single collision domain scenario, the number of interfering players is the *same* for all players. But in a MCD scenario, each node will get a utility which depends on its interference neighbourhood. For example, in Fig. 14, which is got due to execution of the centralized algorithm, players 1, 2, 5, 6 have lower utilities due to their high interference in the network (see Fig. 9), whereas, players 4, 7, 8 have higher utilities due to reduced interference. Similar, behaviour with respect to utility distribution was observed for the other algorithms as well.

Further, we can observe from Figs. 15 and 16 that the utilities obtained from Algorithm 2 is higher than the utilities of players obtained from Algorithm 3 due to the efficiency ratio not converging to 1 for Algorithm 3 which can be attributed to the imperfect information available to it. We also observe from the execution of the three algorithms that, given a fixed number of channels, the utilities of players varies for lower k values and show utility convergence for higher values of k, which can be attributed to fewer NE configurations at higher k values (due to the fact that the contention in the channels is more and hence, every channel tends to have more interference and hence, more congestion in the limited number of channels leading to fewer NE configurations) than at lower k values (due to lesser radios for channel assignment which allows the possibility of different NE configurations in each of the algorithms leading to variation in utilities of the players).

Also, we can observe that there is a variation in the utility obtained by the players due to the three algorithms. This can be seen from Figs. 15 and 16 from the point of view of Player 2 and Player 9 respectively. In Fig. 15, we can see that the utility of Player 2 is higher in the centralized algorithm than the distributed algorithms, whereas we can observe that the distributed algorithms outperform the centralized algorithm in terms of utility to Player 9 (as shown in Fig. 16). Though the centralized algorithm may outperform the distributed algorithms in some cases, the assumptions about global coordination (in centralized algorithm) may not be practical and hence, although there is a trade-off of utility over flexibility, distributed algorithms provide an alternative for achieving reasonably efficient channel assignment schemes.

We plot the payoffs of the players as a scatter plot (Figs. 17, 18) where the results have been accumulated over multiple network configurations. We present the results for Algorithm 3 and for $k = 3, 6$ only due to space constraints. The simulation was performed for other algorithms and other values of k as well. We briefly explain the experimental setup below and then summarize the observations.

Experimental Setup We fixed the number of nodes in the graph as $n = 10$. Generate varying density graphs with edge count = $(n \times t/2)$ corresponding to parameter t where $(1 < t < (n - 1))$. For $t = n - 1$, it denotes a complete graph. 100 random graphs were generated for each value of t. A random graph was drawn from the generated graphs and Algorithm 3 was executed for that graph. The corresponding players' utilities were collected. This process was repeated for 200 times. Finally all the payoff values of all players for all observations were plotted as a scatter plot. We note here that an important parameter is k which is the number of radios per device. We varied value of k from 3 to 6 and observed the corresponding scatter plots for all the three algorithms.

Observations It can be observed from the plots that k is an important parameter in determining the pattern of payoff values for any player. This holds true for all the algorithms considered which can be justified as follows. For low k values (for example, $k = 3$ as shown in Fig. 17), the total number of radios to be allocated a channel in any particular collision domain is smaller and hence the average payoff values are more on the higher side even when the collision domain size is quite high. As k value increases (for example, $k = 6$ as shown in Fig. 18), more and more radios compete for channel allocation leading to lesser player utilities as can be clearly observed from the plots. Another degree of freedom is the size of the user's collision domain. It is observed that as the collision domain size increases, the average utilities of the players are found mostly on the lower values which can be attributed to more radios being allocated to any channel leading to lower utilities for individual radios.

Figure 19 presents the plot of average payoffs per player (over multiple network realizations) as a function of the number of radios for the three algorithms. However, due to space constraints, we present the results for Player 9 only. We briefly explain the experimental setup and the corresponding observations below.

Experimental Setup Sample networks were generated as above. For a particular random graph, all the three algorithms were executed once for each value of k from 3 to 6. The average payoffs of each of the players were collected and then the experiment was repeated for another random graph. This was done for 200 different random networks. Then the average payoffs for the players was calculated for each value of k and the bar charts corresponding to each player was plotted.

Observations In all the algorithms, lower k values lead to more utilities for any player due to number of contending radios being small. For a particular player, Algorithm 2 may lead to better *average* payoff than centralized algorithm for lower k values. For higher k values, there does not seem much difference between the two algorithms. Also, Algorithm 3 (partial information) leads to lower average payoffs than the complete information scenario. In effect, distributed algorithms offer better flexibility and also may lead to finding better NE solutions than their centralized counterpart. Note that the payoff values are all averages over multiple runs with different network configurations. So, there may be cases when centralized outperforms the distributed algorithms in some scenarios and vice versa in others. It is to be noted here that we have shown here the average case behaviour of the algorithms.

7.3.3 Convergence analysis

We now focus on the convergence aspects of the distributed algorithms. Algorithm 2 exhibits good Ω value as shown in Fig. 12. Algorithm 3, on the other hand, has Ω ranging from 0.75 to about 0.9. This lower value of Ω can be attributed to the heuristic nature of the algorithm as, for a player i, the algorithm probabilistically selects channels outside C_i whenever the bound value α becomes higher than the value derived in Eq. 15. To understand this better, we plot the maximum value of ω reached for different values of k. We can see that the maximum ω is between 0.85 and 0.95 for different values of k (see Fig. 20). We correspondingly measure the convergence time needed to achieve this ω value. We note that it takes 25–40 rounds to achieve the maximum Ω as shown in Fig. 21.

7.3.4 Performance evaluation for different networks

In this section, we generate different possible networks and study the behaviour of Algorithm 2 and Algorithm 3 on these diverse networks. As the number of networks possible (with a fixed number of communication sessions i.e., $|T| = N = 10$) is combinatorial in nature, we generate different networks in two ways—(1) Network-based characterization; (2) Conflict graph-based characterization. We should note here that all the results are plotted with 95% confidence interval on the mean values obtained from different generated networks.

Network-based characterization In this type of characterization, the basic idea is to generate different possible MCD networks based on a parameter—Interference Radius, *IR*. Suppose the set of m communication sessions in the network are represented by the set $T = \{(s_i, d_i)\}$ where $i \in \{1, 2, \ldots, m\}$. Now, the interference characteristics of the network are defined through the parameter *IR* as follows: a communication session (s_i, d_i) interferes only with the set of communication sessions in $G = \{(s_j, d_j)\}$ where $j \in \{1, 2, \ldots, m\}$ and $(i - IR) \leq j \leq (i + IR)$ }. By varying IR, we can generate different networks and we studied Ω for Algorithm 2 and Algorithm 3 for different values of *IR* as shown in the Figs. 22 and 23, respectively. Ω converges to 1 for Algorithm 2 while ranges from 0.75 to 0.85 for the Algorithm 3. We can observe that these results are similar to those obtained for the example in Fig. 9.

Conflict graph based characterization The basic idea here is to follow the *conflict graph* model of networks as explained in Sect. 3 to generate different MCD networks. As a result, the problem of generating MCD networks with different interference characteristics can be reduced to the generation of *random* conflict graphs. We used Donald Knuth's Stanford GraphBase (SGB) [29] platform, which is very widely adopted in the literature (e.g., [30]), for generating different network topologies for testing the channel assignment heuristics. The SGB package defines a file-based topology format as well as data structures for representing networks.

Before we present the results obtained by us, let us understand some of the parameters used in the random conflict graph generation process. The random graphs generated are characterized based on (1) number of vertices, (2) number of edges, (3) in-degree distribution of the vertices, (4) out-degree distribution of the vertices and (5) system-independent seed value. We mapped the above parameters to the interference characteristics of the network that we want to test. We assumed the degree distribution of the generated random graph to follow a uniform distribution. We characterize a parameter, threshold average degree (denoted by $thresh_{avg}$), for generating different types of conflict graphs. We set $N = 10$. Each type of conflict graph has a different number of edges denoted by m. We set $m = ((N \times thresh_{avg})/2)$ where $1 \leq thresh_{avg} \leq (N - 1)$. A lower value of $thresh_{avg}$ is indicative of a conflict graph with less interference among the links of the network. This represents a case where each node has a sparse interference in its collision domain. Thus, we generate 100 random graphs for each $thresh_{avg}$ value and then run the distributed algorithms with these random graphs as inputs. In any run, for a given $thresh_{avg}$, a graph is chosen uniformly from the 100 sample graphs generated. The plots for Ω are provided in Figs. 24 and 25. We can observe that the efficiency of Algorithm 2 converges to 1 while Algorithm 3 also achieves reasonably good efficiency (0.75–0.95) under different networks with varied interference neighbourhoods. So, we can now conclude that the behaviour of the

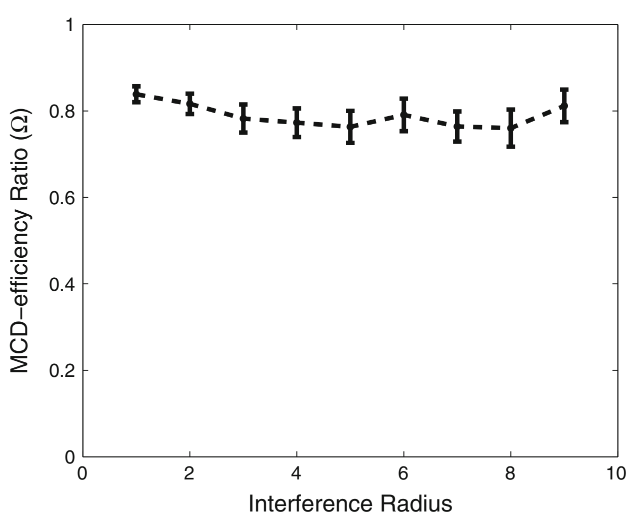


Fig. 23 Interference Radius (IR) versus MCD-efficiency ratio (Ω) for Algorithm 3 with $N = 10$, $|C| = 8$ and $k = 3$

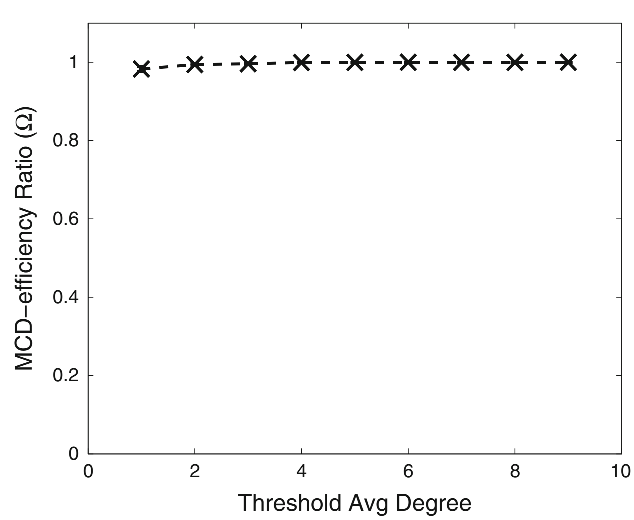


Fig. 24 Threshold avg. degree (t) versus MCD-efficiency ratio (Ω) for Algorithm 2 with $N = 10$, $k = 3$ and $|C| = 8$

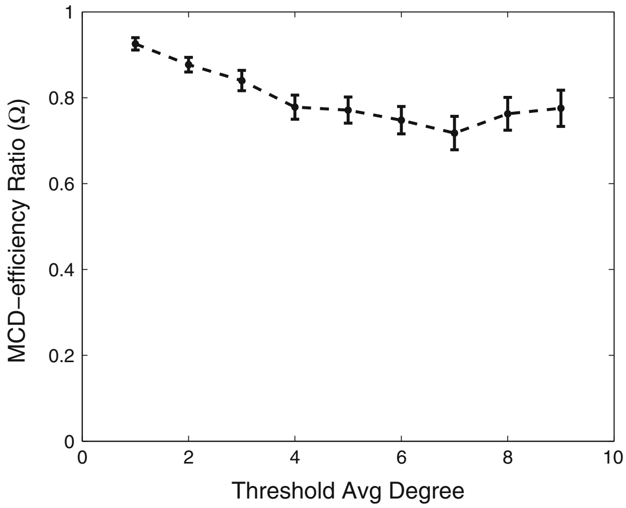


Fig. 25 Threshold avg. degree (t) versus MCD-efficiency ratio (Ω) for Algorithm 3 with $N = 10$, $|C| = 8$ and $k = 3$

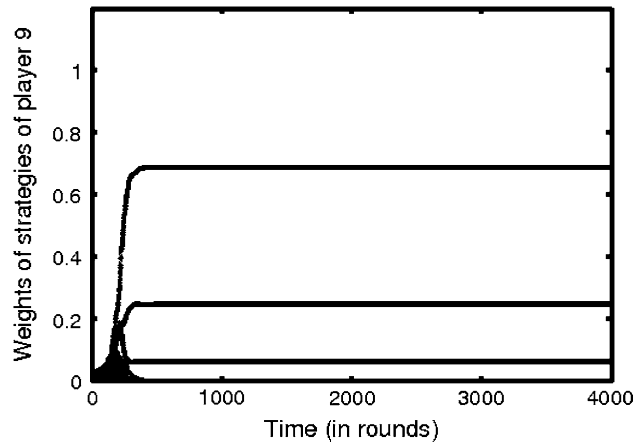


Fig. 26 Probabilities of strategies of player 9 versus time for$N = 10$, $k = 3$

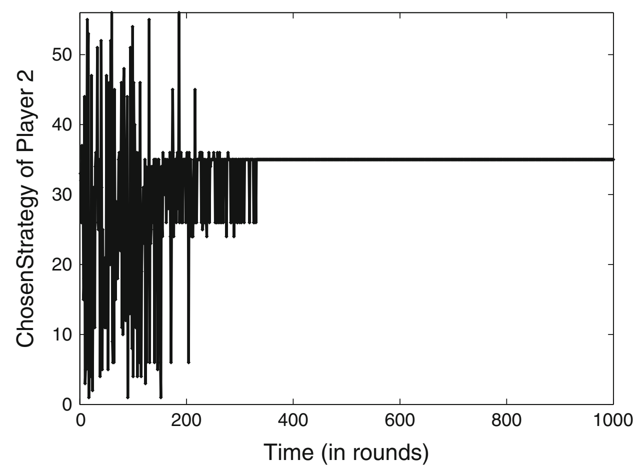


Fig. 27 Strategy evolution of player 2 versus time for $N = 10$ and $k = 3$

distributed algorithms is understood under different networks with varied interference characteristics.

7.3.5 *Observations for no-regret learning scenario*

We will investigate the behaviour of *Hedge* (explained in Sect. 6.4), a distributed adaptation (learning) algorithm based on *regret* minimization which attempts to learn the highest rewarding strategy for a player based on its understanding of the network environment. We will apply the *Hedge* algorithm and try to understand through simulations if it can attain stabilizing outcomes for the MCMR-MCD channel assignment game. *Hedge* was implemented in MATLAB and executed for 4,000 rounds (or iterations) with the example conflict graph given in Fig. 9 as input. We will now summarize the simulation results.

We show the evolution of weights of strategies for *two* players (due to space limitations) which are arbitrarily chosen (i.e. players 2 and 9) in Figs. 26 and 27. We note here that all other players also evolve similarly as these two players. As $|C| = 8$ and $k = 3$, each of the players have ${}^8C_3 = 56$ possible strategies that can be applied in every round. The players start with equal weights for all strategies and as time progresses, they *stabilize* to fixed strategies which yield maximum rewards/utilities. As seen from Fig. 27, player 2 stabilizes to a pure strategy (strategy 35) after about 400 rounds. It can also been seen from Fig. 26 that player 9 stabilizes to multiple strategies (strategies 48, 17 and 14 with probabilities 0.689, 0.248 and 0.063, respectively) after about 400 rounds.

Figure 28 plots the fraction of time during the execution of *Hedge* that the channel assignment is in NE. We verify the NE state by using the conditions derived in Theorem 1. As we can observe, for any value of k, *Hedge* seems to stabilize to outcomes that obey NE conditions of Theorem 1 more frequently as time progresses. This is a significant observation and is conformant with such observations made in works like Greenwald and Jafari [31] and Lim et al. [32] which showed no-regret learning has the potential to converge to pure/mixed/correlated strategy Nash equilibria. It should be however noted that, though there is a possibility of convergence (shown through simulations) to NE by *Hedge*, this requires a more thorough understanding of the connection between no-regret learning and NE. Detailed theoretical examination of the interesting topic of convergence to NE of learning algorithms is planned to be a part of future work of the authors.

Another observation from Fig. 28 is that players with lower values of k learn to play stable outcomes faster. Further, as k value increases, players spend lesser time in stable states which can be attributed to the increase in the

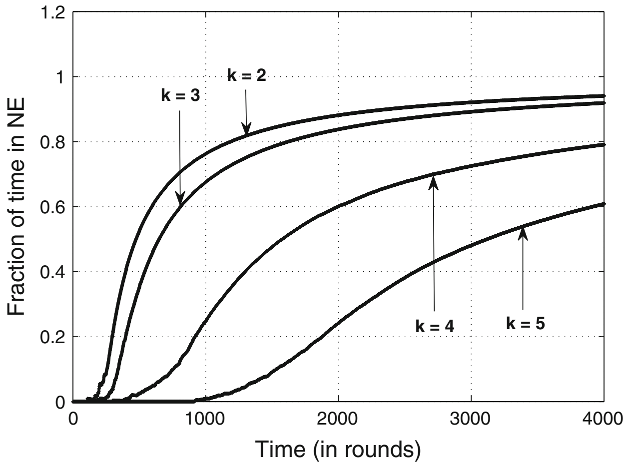


Fig. 28 Percentage of time in NE versus time for $N = 10$ and $|C| = 8$

radios contending for the wireless spectrum resulting in lesser number of stable configurations for convergence.

The collective performance of the players can be seen by observing the behaviour of MCD-efficiency (ω) metric. We plot ω versus time for *Hedge* in Fig. 29. We observe that ω converges to 1 at around 500 rounds which is roughly the time when the strategies of the players stabilize.

We plot the mean of the total utility acquired by the players during the execution of *Hedge* in Fig. 30. We first note that as learning progresses, the mean total utility converges to a fixed value. We note that the mean total utility decreases with increase in k. For lower values of k, the total mean utility is higher due to lesser radios which leads to lower interference.

Thus, it can be briefly summarized that use of reinforcement learning, and in particular, no-regret learning based algorithms in the channel assignment game with multiple collision domain has good potential to give deep insights into the problem.

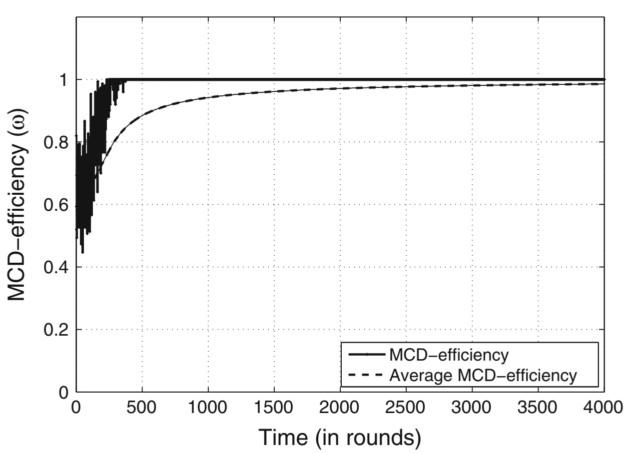


Fig. 29 MCD-efficiency (ω) versus time for $N = 10$, $|C| = 8$, and $k = 3$

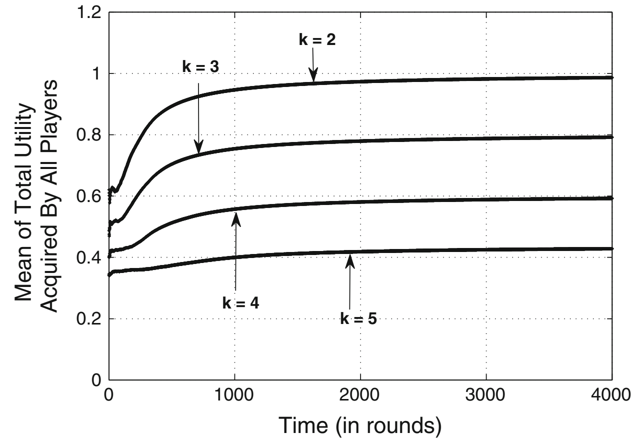


Fig. 30 Mean total utility of all players versus time for $N = 10$ and $|C| = 8$

8 Conclusion and future work

Radio resource management is a critical factor in determining the performance of a wireless network. An important aspect in this complex area is the frequency channel selection for the radio interfaces in the network. A good channel selection algorithm can enable spatial reuse of available wireless channels resulting in reduced interference and corresponding increase in the network capacity.

In this paper, we considered the problem of channel assignment in MCMR networks. We addressed the problem in a MCD context where each link may have different links interfering with it. Any channel assignment algorithm should try to minimize interference experienced by individual links and hence, the problem was formulated as a non-cooperative game where each link will behave in a rational manner and aim to increase its individual throughput. Necessary and sufficient conditions were derived for the network to converge to an NE and then, analysis on the efficiency of the NE was done by deriving the lower bound on the price of anarchy. A new measure of fairness in multiple collision domain context was proposed and detailed fairness analysis was presented.

From the algorithmic perspective of the channel assignment problem, a centralized and two distributed algorithms were proposed based on perfect/imperfect information about the number of interfering radios assigned in a channel and detailed performance evaluation of these algorithms was done based on metrics like efficiency and utility distribution. The algorithms were also shown to exhibit good convergence properties. Experiments were extended to networks with different interference characteristics by using techniques like random conflict graph generation and performance metrics were studied for these networks. Lastly, a no-regret learning based algorithm known as *Hedge* was proposed and behaviour of this algorithm in the channel assignment game was studied. One of the important observations was that *Hedge* converged to stabilizing strategies for all the players and yielded fixed utilities for them.

With respect to future work, the following directions may be interesting and thought-provoking to explore. NE channel assignment schemes may become more interesting if multiple radios of the same player are allowed to be allocated the same channel. Also, game-theoretic analysis of the multiple collision domain setup in a dynamic wireless network may be interesting direction to pursue research. Another direction of work may be to consider the scenario where nodes form coalitions among themselves and act in a cooperative manner. Equilibrium properties can be studied when nodes form coalitions of arbitrary sizes and heuristics can be developed to converge to such an equilibrium if it exists. Also, we considered algorithms which followed a round-based approach. Other learning-based approaches can be thought which asynchronously update the weights for the strategies. The behaviour of such algorithms in a dynamic environment, where player population and their utility structure varies, can be interesting directions for future work. Through this paper, the authors have taken a first step in the direction of casting the channel assignment problem under the operation of a no-regret learning algorithm. But much remains to be explored in this fascinating area. Excellent works like Cesa-Bianchi and Lugosi [33], Young [34] give a detailed account of important contributions by the research community in the fascinating field of learning in games and it is the intention of the authors to pursue further investigation in this direction as part of their future work.

Acknowledgments The authors would like to thank the anonymous reviewers for their valuable comments and suggestions. This work was supported by the Department of Science and Technology, New Delhi, India. It was carried out by Rohith Dwarakanath Vallam when he was pursuing the Master of Science (by Research) degree at the Department of Computer Science and Engineering (CSE), Indian Institute of Technology, Madras (IITM), Chennai, India. Arun A. Kanagasabapathy contributed initially to this work when he was pursuing Bachelor of Technology (B.Tech) degree at the Department of CSE, IITM.

References

1. *IEEE 802.11b standard*, standards.ieee.org/getieee802/download/802.11b-1999.pdf.
2. *IEEE 802.11a standard*, standards.ieee.org/getieee802/download/802.11a-1999.pdf.
3. Aardal, K. I., van Hoesel, S. P. M., Koster, A. M. C. A., Mannino, C., & Sassano, A. (2003). Models and solution techniques for frequency assignment problems. *4OR: A Quarterly Journal of Operations Research, 1*(4), 261–317.
4. Riihijarvi, J., Petrova, M., Mahonen, P., & de Almeida Barbosa, J. (2006). Performance evaluation of automatic channel assignment mechanism for IEEE 802.11 based on graph colouring. In *Proceedings of the 17th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '06)*, September 2006, pp. 1–5.
5. Leung, K., & Kim, B.-J. (2003). Frequency assignment for IEEE 802.11 wireless networks. In *Proceedings of the 58th IEEE Vehicular Technology Conference (VTC '03)*, vol. 3, Oct 2003, pp. 1422–1426.
6. Mishra, A., Banerjee, S., & Arbaugh, W. (2005). Weighted coloring based channel assignment for WLANs. *ACM SIGMOBILE Mobile Computing and Communications Review, 9*(3), 19–31.
7. Alicherry, M., Bhatia, R., & Li, L. E. (2002). Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks. In *Proceedings of the 11th International Conference on Mobile Computing and Networking (MOBICOM '05)*, Aug 2002, pp. 58–72.
8. Shin, M., Lee, S., & Kim, Y. (2006). Distributed channel assignment for multi-radio wireless networks. In *Proceedings of the 3rd IEEE International Conference on Mobile, Ad Hoc and Sensor Systems (MASS '06)*, October 2006, pp. 417–426.
9. Li, N., Hou, J., & Sha, L. (2003). Design and analysis of an MST-based topology control algorithm. In *Proceedings of the 22nd IEEE International Conference on Computer Communications (INFOCOM '03)*, vol. 3, March 2003, pp. 1702–1712.
10. Das, A., Alazemi, H., Vijaykumar, R., & Roy, S. (2005). Optimization models for fixed channel assignment in wireless mesh networks with multiple radios. In *Proceedings of the 2nd IEEE Communication Society Conference on Sensor and Ad Hoc Communications and Networks (SECON '05)*, September 2005, pp. 463–474.
11. Cagalj, M., Ganeriwal, S., Aad, I., & Hubaux, J.-P. (2005). On selfish behaviour in CSMA/CA networks. In *Proceedings of the 24th IEEE Conference on Computer Communications (INFOCOM '05)*, vol. 4, March 2005, pp. 2513–2524.
12. Konorski, J. (2002). Multiple access in ad-hoc wireless LANs with noncooperative stations. In *Proceedings of the 2nd International Conference on Networking of the IFIP Technical Committee on Communication Systems (TC6)*, May 2002, pp. 1141–1146.
13. MacKenzie A. B., & Wicker, S. B. (2003). Stability of multi-packet slotted aloha with selfish users and perfect information. In *Proceedings of the 22nd IEEE International Conference on Computer Communications (INFOCOM '03)*, vol. 3, April 2003, pp. 1583–1590.
14. Halldorsson, M. M., Halpern, J. Y., & Li, L. E. (2004). On spectrum sharing games. In *Proceedings of the 23rd Annual ACM Symposium on Principles of Distributed Computing (PODC' 04)*, July 2004, pp. 107–114.
15. Wu, F., Zhong, S., & Qiao, C. (2008). Globally optimal channel assignment for non-cooperative wireless networks. In *Proceedings of the 27th International IEEE Conference on Computer Communications (INFOCOM '08)*, April 2008, pp. 1543–1551.
16. Nie, N., & Comaniciu, C. (2005). Adaptive channel allocation spectrum etiquette for cognitive radio networks. In *Proceedings of the 1st IEEE International Symposium on New Frontiers in*

Dynamic Spectrum Access Networks (DySPAN '05), Nov 2005, pp. 269–278.
17. Felegyhazi, M., Cagalj, M., Bidokhti, S. S., & Hubaux, J.-P. (2007). Non-cooperative multi-radio channel allocation in wireless networks. In *Proceedings of the 26th International IEEE Conference on Computer Communications (INFOCOM '07)*, March 2007, pp. 1442–1450.
18. Gao, L., & Wang, X. (2008). A game approach for multi-channel allocation in multi-hop wireless networks. In *Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC '08)*, May 2008, pp. 303–312.
19. Gupta, R., Musacchio, J., & Walranda, J. (2007). Sufficient rate constraints for QoS flows in ad-hoc networks. *Ad Hoc Networks*, vol. 5, no. 4, pp. 429–443, May 2007.
20. Luo, H., Lu, S., & Bhargavan, V. (2000). A new model for packet scheduling in multihop wireless networks. In *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MOBICOM '00)*, August 2000, pp. 76–86.
21. Jain, K., Padhye, J., Padmanabhan, V. N., & Qiu, L. (2003). Impact of interference on multi-hop wireless network performance. In *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking (MOBICOM '03)*, September 2003, pp. 66–80.
22. Osborne M. J., & Rubinstein, A. (1994). *A Course in Game Theory*. Cambridge: MIT Press.
23. Bianchi, G. (2000). Performance analysis of the IEEE 802.11 distributed coordination function. *IEEE Journal on Selected Areas in Communications, 18*(3), 535–547.
24. *IEEE standard for wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications*. (1997). P802.11.
25. Koutsoupias, E., & Papadimitriou, C. H. (1993). Worst-case equilibria. In *Proceedings of the 16th Annual Conference in Theoretical Aspects of Computer Science*, vol. 1563, Lecture Notes in Computer Science, 1999, pp. 404–413.
26. Auer, P., Cesa-Bianchi, N., Freund, Y., & Schapire, R. E. (1995). Gambling in a rigged casino: The adversarial multi-armed bandit problem. In *Proceedings of the 36th Annual Symposium on Foundations of Computer Science (FOCS '95)*, October 1995, pp. 322–331.
27. Littlestone, N., & Warmuth, M. K. (1994). The weighted majority algorithm. *Information and Computation, 108*(2), 212–261.
28. Freund, Y., & Schapire, R. (1995). A decision-theoretic generalization of on-line learning and an application to boosting. In *Proceedings of the 2nd European Conference on Computational Learning Theory (EuroCOLT '95)*, May 1995, pp. 23–37.
29. *The Stanford GraphBase: A Platform for Combinatorial Computing*, http://www-cs-faculty.stanford.edu/knuth/sgb.html.
30. Phillips, G., Shenker, S., & Tangmunarunkit, H. (1999). Scaling of multicast trees: Comments on the Chuang-Sirbu scaling law. In *Proceedings of the ACM Annual Conference of the Special Interest Group on Data Communication (ACM SIGCOMM'99)*, pp. 41–51.
31. Greenwald, A., & Jafari, A. (2003). A class of no-regret algorithms and game-theoretic equilibria. In *Proceedings of the 16th Annual Conference on Learning Theory (COLT 2003)*, August 2003, pp. 1–11.
32. Lim, J. G., Chou, C. T., & Jha, S. (2007). Non-cooperative coexistence of co-located independent wireless mesh networks. In *Proceedings of the 4th IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS 2007)*, October 2007, pp. 1–9.
33. Cesa-Bianchi, N., Lugosi, G. (2006). *Prediction, Learning, and Games*. Cambridge: Cambridge University Press.
34. Young, P. (2004). *Strategic Learning and its limits*. Oxford: Oxford University Press.

Author Biographies

Rohith Dwarakanath Vallam received his Bachelor of Engineering degree in Computer Science and Engineering from Visvesvaraya Technological University (VTU), Karnataka, India. He has worked for three years in the software industry in the domain of Intelligent Networks focussing on developing web services in the Voice over Internet Protocol (VoIP) area. He has completed the Master of Science (by Research) degree from the Department of Computer Science and Engineering, Indian Institute of Technology Madras (IITM), Chennai, India. He is currently pursuing his Ph.D. degree at the Indian Institute of Science (IISc), Bengaluru, India. His research interests include application of optimization models in networks. His current research focus is to understand the connections between game theory, mechanism design and machine learning in network economics related applications.

Arun A. Kanagasabapathy received the Bachelor of Technology degree in Computer Science and Engineering from the Indian Institute of Technology Madras (IITM), Chennai, India in 2008. His research focus in his final year project was on conceptualizing and simulating efficient channel assignment strategies for Wireless Mesh Networks. After graduating from IITM, Arun worked for Google as a software engineer. In this role, he worked on the Google's social networking platform trying to detect and reduce spam on the networks. Currently, he is working for Bloomberg as a software engineer, helping it develop trading systems which are widely adopted by multinational banks. Arun is also a recipient of the prestigious National Talent Search Examination (NTSE) scholarship, offered by NCERT. His current research interests include distributed algorithms, game theoretic applications in wireless networks and development of scalable systems in the world of finance.

C. Siva Ram Murthy received the Ph.D. degree from the Indian Institute of Science (IISc), Bengaluru, India. He has been associated with the Department of Computer Science and Engineering, Indian Institute of Technology Madras (IITM), Chennai, India, since 1988, where he is currently a professor. His research interests include parallel and distributed computing, real-time systems, lightwave networks, and wireless networks. He is a senior member of the IEEE.