Spectrum Sharing in Cognitive Radio Networks

Fan Wang, Marwan Krunz, and Shuguang Cui

Department of Electrical & Computer Engineering

University of Arizona

Tucson, AZ 85721

E-mail:{wangfan,krunz,cui}@ece.arizona.edu

Technical Report
TR-UA-ECE-2007-1

Abstract

A key challenge in operating cognitive radios (CRs) in a self-organizing (ad hoc) network is how to adaptively and efficiently allocate transmission powers and spectrum among CRs according to the surrounding environment. In this paper, we present a novel joint power/channel allocation scheme that uses a distributed pricing strategy to improve the network's performance. In this scheme, the spectrum allocation problem is modeled as a non-cooperative game. A price-based iterative water-filling (PIWF) algorithm is proposed, which allows users to converge to the Nash Equilibrium (NE). This PIWF algorithm can be implemented distributively, with CRs repeatedly negotiating their best transmission powers and spectrum. Simulation results show that the social optimality of the NE solution is dramatically improved with our price-based strategy. Based on the orders by which CRs take actions, we study *sequential* and *parallel* versions of the algorithm. We show that the parallel version converges faster than the sequential version. We then propose a MAC protocol that implements our price-based resource allocation algorithm. The proposed MAC protocol allows multiple CR pairs to first contend through an admission phase, and then to iteratively negotiate their transmission powers and spectrum via control-packet exchanges. Subsequently, CRs proceed concurrently with their data transmissions. Simulations are used to study the performance of our protocol and demonstrate its effectiveness in terms of improving the overall network throughput and reducing the average transmission power.

I. INTRODUCTION

The concept of a *cognitive radio* (CR) has recently triggered great interest within the research community (e.g., [9], [11], and [13]). The term "cognitive radio" was first coined by Mitola [16], who used it to refer to a reconfigurable wireless black-box that intelligently changes its communication variables in response to network and user needs. Mitola's definition, however, does not specify the network architecture needed to support CR functionality. More recently, the FCC [8] suggested referring to any radio with adaptive spectrum awareness as a CR. Specifically, a CR should be able to adapt its transmission parameters to the surrounding environment. Several scenarios are envisioned for operating a cognitive radio network (CRN). In this paper, we focus on the role of CRs in *opportunistic communications*. According to this role, CRs are secondary users that coexist with primary radios (PRs) that are licensed to operate over given frequency bands. PR users do not cooperate with or even provide feedback to CR users. CRs continuously sense the spectrum and exploit its "holes" for opportunistic communications.

One of the main challenges in deploying an opportunistic CRN is how to design an efficient and adaptive channel access scheme that supports dynamic channel selection and power/rate allocation in a distributed (ad hoc network) environment. An efficient design is one that tries to maximize the CRN performance without disturbing PR transmissions. A typical measure of efficiency is the achievable sum-rate of all CR transmissions. It is known that the problem of maximizing the sum-rate over a multi-access interference channel subject to individual power constraints is a non-convex optimization problem [4]. Such a problem becomes even more complicated when we allow multiple CRs to share the same channel, as one must now consider the CR-to-CR interference in addition to the PR-to-CR and CR-to-PR interference.

Several attempts have been made to solve the aforementioned "interference channel problem". One well-known resource allocation scheme, called *iterative water-filling* (IWF), was first proposed in [29], where a non-cooperative game was used to model the spectrum management problem, with each user iteratively maximizing its own rate. This per-user optimization problem is convex and leads to a water-filling solution. For the two-user case, it was shown that the *Nash Equilibrium* (NE) exists and the IWF algorithm converges to the NE under certain conditions. However, this NE is generally not Pareto optimal [22] and may be quite inefficient in terms of the sum-rate metric [4]. This is because in a non-cooperative game, each user tries to maximize its own utility function without considering the overall system performance. A centralized spectrum management scheme was proposed in [4]. This scheme greatly improves the system performance over the IWF scheme by utilizing a centralized spectrum management center (SMC). However, such a

centralized approach cannot be applied to a distributed ad hoc CRN, where none of the users has complete knowledge of the entire CRN to function as the SMC.

Given the above, we are motivated to design a channel/power/rate allocation scheme that overcomes the inefficiency of the classic IWF algorithm and yet can be implemented in a distributed fashion. Specifically, we provide incentives to CR users such that they can reach a more *socially efficient* NE. A commonly used incentive technique in game theory is *pricing* (e.g., [5], [20], and [25]) (a thorough review is provided in [10]). Pricing techniques have previously been implemented in various wireless networks, including cellular networks, ad hoc networks, and peer-to-peer networks (e.g., [25], [5], [28] and [1]). In this paper, we apply pricing techniques to a distributed CRN, and propose a *price-based iterative water-filling* (PIWF) algorithm. We show that this PIWF algorithm maintains the simplicity and distributed operation of the original IWF algorithm; yet, it achieves better bandwidth efficiency in the form of higher sum-rate. The effectiveness of the pricing approach depends on the appropriate selection of the "pricing function," which is a challenging problem by itself. Although there may exist an "optimal" pricing function that allows the NE to converge to the Pareto optimum, the search for such a function generally requires a central controller and is hard to implement in a distributed manner. Some sub-optimal pricing functions were proposed in the literature. For example, the authors in [5] proposed an auction-like pricing scheme for *single-channel* mobile ad hoc networks (MANETs). In this scheme, the unit price (uniform across all users) is gradually increased until the system reaches a feasible NE. A similar approach (also, for a single channel) was used in [25], where the users of a wireless data network keep increasing their prices in a uniform fashion until the utility of one user begins to decrease. Both of the previously mentioned pricing schemes achieve a feasible NE and improve the system performance. However, the achieved NEs are not guaranteed to be globally optimal, which is partially due to the fact that both of the two approaches assume a uniform unit price for all players in the game. In our work, we use a *user-dependent* pricing function, which we found to improve the sum-rate of the achieved NE after a few iterations. Such a pricing function can be determined by allowing each CR user to distributively collect some neighborhood information through the exchange of control packets.

Another challenge in applying the classic IWF algorithm [29] to CRNs is that this algorithm only considers a *total* power constraint on the transmission of each user. In a CRN, PRs impose a strict power constraint over *each* frequency band, so CR transmissions have to abide by a *frequency-dependent* power mask. Such a mask affects the response of each CR user and thus the achieved NE. In this paper, we

incorporate such a frequency-dependent power mask into the optimization problem.

In our proposed algorithm, each user maximizes its own utility function (which includes a pricing term) by performing a single-user price-based water-filling, while treating the interference from other CR users at each sub-band as additive white Gaussian noise (AWGN). The same procedure iterates sequentially, eventually converging to the NE. When the number of users in the network is large, sequential updating of transmission powers can suffer from slow convergence. Therefore, we also study a *parallel* version of the PIWF algorithm (the parallel concept for the IWF algorithm was first introduced in [26]). This parallel algorithm is an instance of the Jacobi scheme: At each iteration, CRs update their strategies simultaneously, based on the measured interference in the previous iteration. Simulations indicate that this parallel version converges faster than the sequential version of the PIWF algorithm.

Both the sequential and parallel PIWF algorithms require CRs to be synchronized and the system parameters to be correctly estimated at each CR. These conditions may not be satisfied in practice. To overcome this problem, a "relaxed" update scheme has been used in the literature (e.g., in [2], [15], and [26]) and is studied in our work for completeness. For the "relaxed" version of the PIWF algorithm, each CR is required to remember its most recent policy choices along with the choices of other users. As such, the relaxed update scheme is more robust to inaccurate estimation and channel oscillations, but it may lead to certain degradation in the convergence speed.

Our PIWF algorithms are then integrated into the design of a distributed medium access (MAC) protocol for CRNs. This protocol allows CRs to dynamically select channels and adapt their transmission powers and rates. We show that the specific implementation of the PIWF algorithm impacts the MAC protocol design. Simulations are conducted to compare the performance of the proposed protocol against other adaptive protocols.

The rest of this paper is organized as follows. The system model is described in Section II. Section III formulates the non-cooperative game and introduces the pricing techniques. We discuss the PIWF algorithms in Section IV and design the corresponding MAC protocol in Section V. In Section VI, we provide simulation results of the PIWF algorithms and compare them with the classic IWF algorithm. Conclusions are drawn in Section VII.

II. SYSTEM MODEL

We consider a hybrid network that consists of several primary radio networks (PRNs) and one CRN. The CRN consists of N CR pairs (links). The total spectrum is composed of K orthogonal frequency

channels ($K < N$) with central frequencies $f_1, f_2, \ldots, f_K$. Each PR user in a PRN may operate over one or multiple channels. The PRs in the network are modelled as an ON/OFF source, where "ON" means that the PR user is actively transmitting.

Each CR user may simultaneously transmit over multiple channels. Similarly, each CR user can receive over multiple channels (from the same transmitter) at the same time. However, we require the operation to be half-duplex, meaning that a CR user cannot receive while transmitting, and vice versa. When not transmitting, a CR user is also capable of measuring the total noise-plus-interference (TNPI) level over each channel. Let $M_i(k)$ denote the TNPI level measured by CR user i over channel k. This quantity includes the received PR-to-CR interference, the CR-to-CR interference, and the thermal noise. Let $\mathbf{M}_i \stackrel{\text{def}}{=} [M_i(1), M_i(2), \ldots, M_i(K)]^1$. This vector is used by CR i to perform dynamic channel selection, power control, and rate allocation, as described later.

Previous work on distributed resource allocation for CRNs (e.g., [27]) assumed that CR transmissions do not interfere with each other, i.e., only one CR link can be active over a given channel in a given neighborhood (along with the PRs). In this case, there is no spectrum overlap among neighboring CR users. Such an approach enjoys simplicity (CSMA/CA-like design) but it limits the number of admitted CR links. In our work, we allow multiple CR users to overlap in the allocated frequency channels. To illustrate, Figure 1 gives a channel allocation example for a CRN with $K = 3$ and $N = 4$. A shaded square means that a channel is utilized by a CR link. For example, link 1 uses channels 1 and 2, while link 4 uses only channel 1. We denote the set of utilized channels for CR link i as S_i. In our example, $S_1 = \{1, 2\}$ and $S_4 = \{1\}$. The transmission power vector of CR link i over various channels is denoted by $\mathbf{P}_i = [P_i(1), P_i(2), \ldots, P_i(K)]$, where $P_i(k)$ is the transmission power of CR i on channel k. If channel $k \in S_i$, then $P_i(k) > 0$; otherwise, $P_i(k) = 0$.
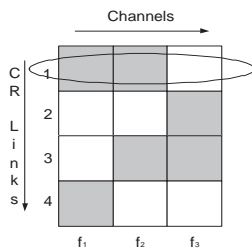

Fig. 1: Example of channel allocation with 4 CR links in the same neighborhood.

We impose the following constraints:

---

[1]Throughout this paper, vector quantities are indicated in bold font.

1) *Maximum transmission power constraint*: The total transmission power of a CR user over the selected channels should not exceed P_{max}, i.e, $\sum_{k \in S_i} P_i(k) \leq P_{max}$. Here, we assume that the total power constraint is the same for all users. It is easy to extend the treatment to the case where P_{max} is user-dependent.

2) *CR-to-PR power mask constraint*: The transmission power of CR i on channel k is constrained by the power mask $P_{mask}(k)$. Let $\mathbf{P}_{mask} \overset{\text{def}}{=} [P_{mask}(1), P_{mask}(2), \ldots, P_{mask}(K)]$ denote the power mask vector. $\mathbf{P}_{mask}$ is set in a way that CR transmissions result in no disturbance to PR transmissions. The determination of $\mathbf{P}_{mask}$ is itself an important issue, but is out of the scope of this paper. In the following analysis, we assume that $\mathbf{P}_{mask}$ is given a priori.

We assume that the CRs are either static or move slowly (relative to the convergence time of the resource assignment algorithm). This assumption is generally acceptable because our iterative algorithms are implemented on the time scale of few milliseconds, whereas pedestrian and vehicular mobility impacts the network topology on the time scale of seconds. In addition, we assume that all the CR nodes follow the same operation rules and have the same system constraints.

III. PROBLEM FORMULATION

In a non-cooperative CRN, each CR user is interested in maximizing its own achievable rate. Such a greedy behavior can be modeled using game theory. Game theory analyzes players' interactions in decision-making processes. It can be used to identify distributed optimal strategies for the players [18] [19]. A normal game can be expressed as $\mathcal{G} = \{\Omega, \mathcal{P}, \{U_i\}\}$, where $\Omega = \{1, 2, \ldots, N\}$ is a finite set of rational players; $\mathcal{P} = \mathcal{P}_1 \times \mathcal{P}_2 \times \ldots \times \mathcal{P}_N$ is the action space with $\mathcal{P}_i$ being the action set for player i; and $U_i \colon \mathcal{P} \to \mathcal{R}$ is the utility (payoff) function of player i, which depends on the strategies of all players. We model the channel/power allocation problem in a CRN as a non-cooperative game, in which the players are the CR users; their actions are the transmission powers (i.e., for user i, its action is given by $\mathbf{P}_i = [P_i(1), P_i(2), \ldots, P_i(K)]$); and their utility functions are associated with their actions and the quality of the channels. Note that a CR user in the game denotes a CR link consisting of a pair of CR nodes.

A. Utility Function

The utility function of CR user i can be considered as the reward received by this user from the network. It depends on user i's action $\mathbf{P}_i$ and the union set of all other users' actions $\mathbf{P}_{-i}$, where

$\mathbf{P}_{-i} \stackrel{\text{def}}{=} [\mathbf{P}_1, \ldots, \mathbf{P}_{i-1}, \mathbf{P}_{i+1}, \ldots, \mathbf{P}_N]^T$. While the selection of the utility function is not unique, this selection must have a physical meaning for the given application. A natural selection of the utility function for CR link i (also used in [6], [26], [29]) is given by:

$$U_i(\mathbf{P}_i) = \sum_{k=1}^{K} u_i(P_i(k)) = \sum_{k=1}^{K} \left[\log_2(1 + \frac{h_{ii}(k)P_i(k)}{\sum_{j \neq i} h_{ji}(k)P_j(k) + M_i^{(PR)}(k) + N_i(k)}) \right] \tag{1}$$

where $h_{ii}(k)$ denotes the channel gain between the transmitter and the receiver of link i over channel k, $h_{ji}(k)$ denotes the channel gain between the transmitter of link j and the receiver of link i over channel k, $M_i^{(PR)}(k)$ is the PR-to-CR interference at the receiver of CR link i over channel k, and $N_i(k)$ is the thermal noise power on channel k.

Given the above utility function, users iteratively select their transmission powers to maximize their own utility functions, and eventually converge to a NE after several iterations (under conditions that will be discussed later). As mentioned before, because of the non-cooperative nature of the game, each CR user behaves selfishly. Thus, the resulting NE may be far from the Pareto optimum, defined as:

$$\begin{aligned} U_{\text{opt}} &\stackrel{\text{def}}{=} \max_{\{\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_N\}} \sum_{i=1}^{N} w_i U_i(\mathbf{P}_i) \\ &= \max_{\{\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_N\}} \sum_{i=1}^{N} w_i \sum_{k=1}^{K} u_i(P_i(k)) \end{aligned} \tag{2}$$

where w_i denotes the "weight" assigned to user i, which may be interpreted in different ways (e.g., priority factor of user i).

To drive the NE towards the above Pareto optimum, we use pricing as an incentive for each CR user. Accordingly, we define a new utility function for user i as follows:

$$\tilde{U}_i(\mathbf{P}_i) = \sum_{k=1}^{K} \tilde{u}_i(P_i(k)) \tag{3}$$

with

$$\tilde{u}_i(P_i(k)) \stackrel{\text{def}}{=} -c_i(k) + \log_2 \left[1 + \frac{h_{ii}(k)P_i(k)}{\sum_{j \neq i} h_{ji}(k)P_j(k) + M_i^{(PR)}(k) + N_i(k)} \right] \tag{4}$$

where $c_i(k)$ represents the pricing function for user i on channel k. As discussed in Section I, our goal is to choose a user-dependent pricing function that can drive the CR users to converge to an efficient NE. How to define this pricing function will be discussed in Section III-C.

B. Game Formulation

Given the price-based utility function in (4), each CR user i iteratively selects its power vector $\mathbf{P}_i$ so as to maximize $\tilde{U}_i(\mathbf{P}_i)$ subject to the constraints listed in Section II. This results in the following non-cooperative game $\mathcal{G}$:

$$
\begin{aligned}
&\underset{\mathbf{P}_i}{\text{maximize}} \quad \tilde{U}_i(\mathbf{P}_i), \quad \forall i \in \{1, 2, \ldots, N\} \\
&\text{s.t.} \\
&\text{C1:} \qquad P_i(k) \geq 0, \quad \forall i \in \{1, 2, \ldots, N\} \text{ and} \\
&\qquad\qquad\qquad \forall k \in \{1, 2, \ldots, K\} \\
&\text{C2:} \qquad \textstyle\sum_{k=1}^{K} P_i(k) \leq P_{max}, \quad \forall i \in \{1, 2, \ldots, N\} \\
&\text{C3:} \qquad P_i(k) \leq P_{mask}(k), \quad \forall i \in \{1, 2, \ldots, N\} \\
&\qquad\qquad\qquad \text{and } \forall k \in \{1, 2, \ldots, K\}.
\end{aligned}
\tag{5}
$$

If $P_{max} \geq \sum_k P_{mask}(k)$, then constraint C2 can be omitted. If there is a solution to the above game, then it is a one that achieves a NE. Note that the above game differs from the game studied in [29] in the form of the utility function and in the addition of the power mask constraint. Thus, the NE existence proofs in [6] and [29] cannot be directly applied here. However, from the following proposition, we show that a NE solution always exists for the above game.

Proposition 1: For any given P_{max} and $\mathbf{P}_{mask}$ values, there is at least one NE for the game $\mathcal{G}$ in (5).

Proof: the game in our setup can be shown to be a concave game if the following two properties are satisfied:

1) The action space $\mathcal{P}$ is a closed and bounded convex set;

2) The utility function $\tilde{U}_i(\mathbf{P}_i)$ is concave over its strategy set.

Because a concave game always admits at least one NE [24], we have the above proposition. ∎

Given the existence of a NE solution, we need to design an algorithm for CR users to reach this NE. We address this issue in the forthcoming sections. However, before we do that, we first discuss the optimal pricing function.

C. Optimal Pricing Function

To illustrate the impact of pricing on improving the efficiency of the NE, in Figure 2, we depict an example of the Pareto optimal frontier and the NE for a two-user game. In general, the NE is not Pareto

optimal. One contribution of our work is in introducing a user-dependent linear pricing function that drives the NE close to the Pareto optimal frontier with each player using only the neighborhood information about the network. As explained in Section V, this information is acquired via control-packet exchanges during the channel access process.
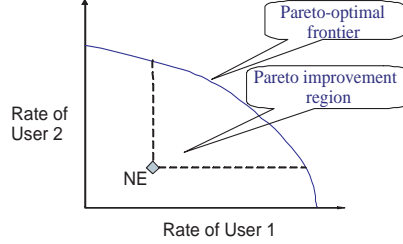

Fig. 2: Nash equilibrium and Pareto-optimal frontier for a 2-user game.

Proposition 2: Consider the game $\mathcal{G}$ with utility function $\tilde{U}_i$, $i = 1, 2, \ldots, N$, as defined in (3), and let the pricing function $c_i(k)$ be given by $c_i(k) = \lambda_i(k)P_i(k)$. Then, the game has at least one NE solution (from Proposition 1). Further, if this NE solution is Pareto optimal, then the pricing factor $\lambda_i(k)$ must be of the form:

$$\lambda_i(k) = \frac{1}{w_i} \sum_{j \in \mathbf{NBR}_i} w_j \Big[\frac{h_{jj}(k)P_j(k)h_{ij}(k)}{M_j(k)(M_j(k) + h_{jj}(k)P_j(k))} \Big] \tag{6}$$

where $\mathbf{NBR}_i$ denotes the set of neighbors of CR user i.

Proof: By definition, a NE is the solution to the individual utility optimization problem for each user given all other users' actions. In our formulation, each individual optimization problem is a convex problem with the linear constraints C1-C3 in (5). So the Lagrangian function for user i can be written as:

$$\begin{aligned}
J_i &= w_i \sum_k \tilde{u}_i(P_i(k)) + \sum_k \alpha_{i,k} P_i(k) \\
&\quad - \beta_i \Big[\sum_k P_i(k) - P_{max} \Big] - \sum_k \gamma_{i,k} \big[P_i(k) - P_{mask}(k) \big] \\
&= w_i \sum_k \big[u_i(P_i(k)) - \lambda_i(k)P_i(k) \big] + \sum_k \alpha_{i,k} P_i(k) \\
&\quad - \beta_i \big(\sum_k P_i(k) - P_{max} \big) - \sum_k \gamma_{i,k}(P_i(k) - P_{mask}(k))
\end{aligned}$$

where $\alpha_{i,k}$, β_i, and $\gamma_{i,k}$ are the Lagrangian multipliers (non-negative real numbers). The Karush-Kuhn-Tucker (KKT) conditions [3] for user i are given by:

$$w_i \frac{\partial u_i(P_i(k))}{\partial P_i(k)} - w_i \lambda_i(k) + \alpha_{i,k} - \beta_i - \gamma_{i,k} = 0, \forall k$$

$$p_i(k) \geq 0, \forall k$$

$$\alpha_{i,k} p_i(k) = 0, \forall k$$

$$\sum_k P_i(k) - P_{max} \leq 0 \tag{7}$$

$$\beta_i \Big[\sum_k P_i(k) - P_{max} \Big] = 0$$

$$P_i(k) - P_{mask}(k) \leq 0, \forall k$$

$$\gamma_{i,k}(P_i(k) - P_{mask}(k)) = 0, \forall k$$

On the other hand, to solve the social optimization problem (2) with constraints C1-C3, the Lagrangian function can be written as:

$$
\begin{aligned}
J &= \sum_{i=1}^{N} w_i \sum_{k=1}^{K} u_i(P_i(k)) + \sum_{i=1}^{N} \sum_{k=1}^{K} \alpha_{i,k} P_i(k) \\
&\quad - \beta_i \sum_{k=1}^{K} \big[(P_i(k) - P_{max}(k)) \big] \\
&\quad - \sum_{i=1}^{N} \sum_{k=1}^{K} \gamma_{i,k} \big[P_i(k) - P_{mask}(k) \big] \\
&= w_i \sum_{k=1}^{K} u_i(P_i(k)) + \sum_{j=1, j \neq i}^{N} w_j \sum_{k=1}^{K} u_j(P_j(k)) \\
&\quad + \sum_{i=1}^{N} \sum_{k=1}^{K} \alpha_{i,k} P_i(k) - \beta_i \sum_{k=1}^{K} [P_i(k) - P_{max}(k)] \\
&\quad - \sum_{i=1}^{N} \sum_{k=1}^{K} \gamma_{i,k} [P_i(k) - P_{mask}(k)]
\end{aligned}
$$

In this case, the KKT conditions are given by:

$$w_i \frac{\partial u_i(P_i(k))}{\partial P_i(k)} + \sum_{j=1, j\neq i}^{N} w_j \frac{\partial u_j(P_j(k))}{P_i(k)}$$

$$+\alpha_{i,k} - \beta_i - \gamma_{i,k} = 0, \forall i, k$$

$$p_i(k) \geq 0, \forall i, k$$

$$\alpha_{i,k} p_i(k) = 0, \forall i, k$$

$$\sum_{k=1}^{K} P_i(k) - P_{max} \leq 0, \forall i \tag{8}$$

$$\beta_i \Big[\sum_{k=1}^{K} P_i(k) - P_{max} \Big] = 0, \forall i$$

$$P_i(k) - P_{mask}(k) \leq 0, \forall i, k$$

$$\gamma_{i,k}(P_i(k) - P_{mask}(k)) = 0, \forall i, k$$

To obtain the same solution to the two sets of KKT conditions (7) and (8), we must have:

$$\lambda_i(k) = -\frac{1}{w_i} \sum_{j\neq i} w_j \frac{\partial u_j(P_j(k))}{\partial P_i(k)}. \tag{9}$$

By substituting $u_j(k)$ into (9), we have:

$$\lambda_i(k) = \frac{1}{w_i} \sum_{j\neq i} w_j \Big[\frac{h_{jj}(k)P_j(k)h_{ij}(k)}{M_j(k)(M_j(k) + h_{jj}(k)P_j(k))} \Big]. \tag{10}$$

If the transmitter of link i and the receiver of link j are not neighbors, i.e., the transmission of link i at the maximum power cannot reach the receiver of link j, the channel gain $h_{ij}(k)$ is set to zero. Thus, the optimal pricing factor for link i only depends on its neighborhood information. We then have the result in Proposition 2. ∎

Intuitively, a higher pricing factor $\lambda_i(k)$ will prevent user i from using a large transmission power on channel k. In view of (6), for link i to determine its optimal pricing factor, the following procedure is needed: If a neighbor j is to transmit over channel k, it needs to broadcast its transmission power $P_j(k)$, the measured TIPN $M_j(k)$, and the channel gain $h_{jj}(k)$. The above information will be conveyed using MAC control packets (details in Section V). In addition, the channel gains $h_{ij}(k)$ can be computed by measuring the received signal power of the MAC control packets.

IV. ITERATIVE ALGORITHMS

From the results of the previous section, we can use the following iterative algorithm to reach a NE for the game $\mathcal{G}$: Each CR user, say i, first adjusts its linear pricing factor $\lambda_i(k)$ over all channels according to (6), and then determines its *best response*, i.e., the optimal channel/power combination based on the measured $\mathbf{M}_i$. The best response of user i is to maximize its individual utility function (4) subject to the constraints C1-C3. The same procedure is repeated for all users in the network. If such a procedure converges, then by definition, it has to converge to a NE of the game in (5).

Note that for user i, its utility function in (1) is monotonically increasing in $P_i(k)$ given that the other users' powers are fixed. So the only factor that prevents user i from choosing infinitely large transmission powers is constraint C2. In our work, after adding the linear pricing function, the utility function (4) now leads to a finite optimal power setting even without constraint C2.

Proposition 3: By treating the other users' transmissions as interference, the best response of user i is given by:

$$\mathbf{P}_i^* = \mathbf{BR}_i(\mathbf{P}_{-i}) = [BR_i(\mathbf{P}_{-i}(1)), \dots, BR_i(\mathbf{P}_{-i}(K))] \tag{11}$$

where

$$BR_i(\mathbf{P}_{-i}(k)) \stackrel{\text{def}}{=} \left[\frac{1}{\beta + \lambda_i(k)} - \frac{M_i(k)}{h_{ii}(k)}\right]_0^{P_{mask}(k)} \tag{12}$$

where the function $[x]_a^b$, with $b > a$, denotes the Euclidean projection of x onto the interval $[a, b]$, i.e., $[x]_a^b = a$ if $x < a$, $[x]_a^b = x$ if $a < x < b$, and $[x]_a^b = b$ if $x > b$. The parameter β is known as the water level. It is determined by user i as the minimum non-negative value that results in satisfying the total power constraint C2.

Proof: We first solve the optimization problem without the power mask constraint C3, using the method of Lagrange multipliers. This leads to a water-filling solution of the form:

$$P_i^*(k) = [\frac{1}{\beta + \lambda_i(k)} - \frac{M_i(k)}{h_{ii}(k)}]^+, i = 1, 2, \dots, N. \tag{13}$$

If $P_i^*(k)$ is the optimal power allocation over channel k, then the slope of the utility function $u_i(P_i(k))$ must be positive at the point $P_i^*(k)$. Otherwise, a power vector $\mathbf{P}_i$ with a smaller $P_i(k)$ could reach a higher utility $U_i(P_i)$, with all the constraints satisfied. Thus, the utility function $u_i(P_i(k))$ is monotonically increasing between 0 and $P_i^*(k)$. Due to this monotonicity, if any of the $P_i^*(k)$ in (13) violates the upper bound C3, then the corresponding bounded optimal solution must be the upper bound $P_{mask}(k)$ itself (a

similar approach was also adopted in [7]). After bounding the $P_i^*(k)$ by $P_{mask}(k)$, the remaining power will be further water-filled over other channels, thus reaching the result in (11).

A similar result for the IWF algorithm is provided in [26]. Although we have additional pricing function, a similar analysis can be used to reach the same result in Proposition 3. ■

Note that without the power mask constraint (i.e., $P_{mask}(k) = +\infty$ for all k) and without the pricing function (i.e., $\lambda_i(k) = 0$ for all k and i), (11) and (12) represent the classic water-filling solution. Figure 3 graphically illustrates the difference between the traditional water-filling [29] and the price-based water-filling solution in (12).
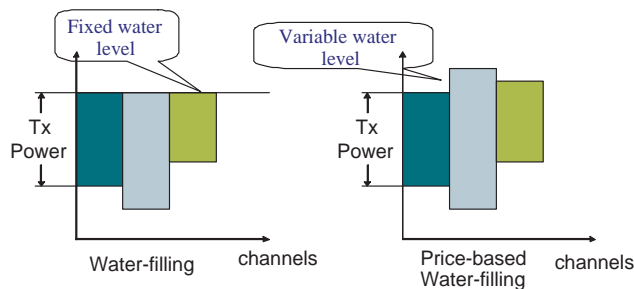

Fig. 3: Classic water-filling versus price-based water-filling.

Several approaches can be used by CR users to reach the NE according to the best-response function in (11). CR users may make their decisions one after another or in parallel, which corresponds to a sequential or a parallel update procedure. Next, we describe these two procedures and analyze their convergence properties.

A. Sequential Price-based Iterative Water Filling

In the sequential PIWF algorithm, CR users implement their best-response decisions sequentially, according to some given order. Let $\mathbf{P}_i^{(l)}$ be the power vector of user i in the lth iteration. The condition $\frac{||\mathbf{P}_i^{(l)} - \mathbf{P}_i^{(l-1)}||}{||\mathbf{P}_i^{(l-1)}||} \leq \varepsilon$ is used as the stopping criteria for the PIWF algorithm, where ε is a small number (e.g., 0.01). If this condition is not satisfied within a certain number of iterations (L_{max}), the algorithm will also stop. The above algorithm is akin to the *Gauss-Seidel* procedure, where the players take their turns sequentially and act on the most recent policy information obtained from other players. A pseudo-code for the sequential PIWF algorithm is depicted in Algorithm 1.

In a two-user scenario, $\mathbf{P}_1^{(l+1)}$ for user 1 can be expressed as:

$$\mathbf{P}_1^{(l+1)} = \mathbf{BR}_1(\mathbf{BR}_2(\mathbf{P}_1^{(l)})) \stackrel{\text{def}}{=} \mathbf{T}(\mathbf{P}_1^{(l)}) \tag{14}$$

Algorithm 1 Sequential PIWF

Initialize $P_i(k) = 0, \forall i$ and k.
Initialize iteration count $l = 0$.
Repeat:

  1: $l = l + 1$;
  2: **for** $i = 1$ to N users **do**
  3:     **for** $k = 1$ to K channels **do**
  4:         Estimate the TIPN level $M_i(k)$;
  5:         Compute the pricing factor $\lambda_i(k)$ using (6);
  6:         Estimate the channel gain $h_{ii}(k)$ using the received signal power of the control packet.
  7:     **end for**
  8:     $\mathbf{P}_i^{(l)} = \mathbf{BR}_i(\mathbf{P}_1^{(l)}, \ldots, \mathbf{P}_{i-1}^{(l)}, \mathbf{P}_{i+1}^{(l-1)}, \ldots, \mathbf{P}_N^{(l-1)})$;
  9:     Transmit on selected channels using $\mathbf{P}_i^{(l)}$.
10: **end for**

Until $l > L_{max}$ or $\frac{||\mathbf{P}_i^{(l)} - \mathbf{P}_i^{(l-1)}||}{||\mathbf{P}_i^{(l-1)}||} \leq \varepsilon$ for all i.

---

where $\mathbf{T}$ is a vector of concatenated functions, given by $\mathbf{BR}_1(\mathbf{BR}_2(.))$. For a general N-user game, the expression is more complicated. For simplicity, we keep the notation $\mathbf{T}$ as the mapping between the previous power vector and the current power vector. To ensure convergence to the NE for the classic IWF algorithm, several sufficient conditions have been proposed in the literature. The convergence condition was first provided in [29] for the two-user case and in [6] for the N-user case. More recently, the convergence conditions were further relaxed in [12] and [26].

Because the utility function in (4) includes a pricing term that is not present in the formulations in [12] and [26], the convergence proofs of [12] and [26] cannot be applied here. In fact, because the pricing factor $\lambda_i(k)$ is recomputed in every iteration (as shown in Algorithm 1), the mapping $\mathbf{T}$ is time-varying over iterations. Thus the fixed-point theorem that underlies the proofs in [12] and [26] cannot be applied. The convergence proof under a time-varying mapping function is a challenging problem and will be left for future work. However, convergence has always been observed in our simulations. Figure 4 depicts the convergence behavior over several iterations with $N = 10$ and $K = 5$. The figure shows the average sum-rate improvement of the sequential PIWF over the classic IWF algorithm for 1000 runs, with the starting sum-rate of the IWF algorithm normalized to one. The two algorithms converge at comparable speeds, but the NE solution for the sequential PIWF algorithm is much better than the NE of the classic IWF algorithm.

Although the convergence proof for a time-varying pricing function is difficult to establish, if the pricing factor remains fixed over several iterations, the convergence proof in [26] is still applicable. This is because
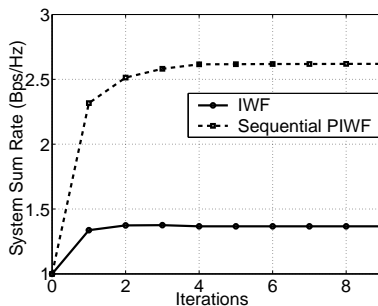
Fig. 4: Normalized sum-rate versus iterations (starting value for the sum-rate of the classic IWF is normalized to one).

adding a linear pricing function with a fixed pricing factor to the utility function (1) has no impact on the convergence proof in [26]. If we apply the analysis in [26] to our CRN formulation in Section III, we have the following proposition:

Proposition 4: Suppose that the pricing function in (4) takes a linear form with a fixed pricing factor over a few iterations. Then, the sequential update procedure converges to the unique NE if one of the following two sets of conditions is satisfied:

$$(1) \quad \sum_{j=1,j\neq i}^{N} \max_{k \in S_i \cap S_j} \frac{h_{ji}(k)}{h_{ii}(k)} < 1, \forall i \in 1, 2, \ldots, N \tag{15}$$

$$(2) \quad \sum_{i=1,i\neq j}^{N} \max_{k \in S_i \cap S_j} \frac{h_{ji}(k)}{h_{ii}(k)} < 1, \forall j \in 1, 2, \ldots, N. \tag{16}$$

From (15) and (16), the convergence and the uniqueness of NE are ensured if the CRs that share the same channel are far apart, and thus inflict weak interference on each other.

When the number of CR users in the network is large, the sequential update approach may suffer from slow convergence speed. Therefore, we present a *parallel* PIWF version that is suitable for dense CRNs.

B. Parallel Price-based Iterative Water Filling

In the parallel PIWF algorithm, the CR users act in parallel, instead of sequentially. The stopping criteria are the same as those of the sequential PIWF. The algorithm is described in Algorithm 2. This algorithm is related to the *Jacobi* computational procedure, where CR users simultaneously perform price-based water-filling in each iteration, based on the interference generated by other users in the previous iterations. In the two-user case, the counterpart of (14) is:

$$\mathbf{P}_1^{(l+2)} = \mathbf{BR}_1(\mathbf{P}_2^{(l+1)}) = \mathbf{BR}_1(\mathbf{BR}_2(\mathbf{P}_1^{(l)})) = \mathbf{T}(\mathbf{P}_1^{(l)}). \tag{17}$$

In [26], it was proved that the convergence conditions for the parallel and sequential IWF are the same. The same proof is not applicable if the mapping function $\mathbf{T}(.)$ is time-varying. However, if the pricing factor of the linear pricing function remains fixed, we can apply the corresponding proof and reach the following corollary of Proposition 4.

Algorithm 2 Parallel PIWF

Initialize $P_i(k) = 0, \forall i$ and k.
Initialize iteration count $l = 0$.
Repeat:
  1: $l = l + 1$;
  2: **for** $i = 1$ to N users **do**
  3:     **for** $k = 1$ to K channels **do**
  4:         Estimate the TIPN level $M_i(k)$;
  5:         Compute the pricing factor $\lambda_i(k)$ using (6);
  6:         Estimate the channel gain $h_{ii}(k)$ using the received signal power of the control packet.
  7:     **end for**
  8:     $\mathbf{P}_i^{(l)} = \mathbf{BR}_i(\mathbf{P}_1^{(l-1)}, \ldots, \mathbf{P}_{i-1}^{(l-1)}, \mathbf{P}_{i+1}^{(l-1)}, \ldots, \mathbf{P}_N^{(l-1)})$;
  9: **end for**
  10: **for** $i = 1$ to N users **do**
  11:     Transmit using $\mathbf{P_i}^{(l)}$.
  12: **end for**
Until $l > L_{max}$ or $\frac{||\mathbf{P}_i^{(l)} - \mathbf{P}_i^{(l-1)}||}{||\mathbf{P}_i^{(l-1)}||} \leq \varepsilon$ for all i.

Corollary 1: If the conditions in Proposition 4 are satisfied, the parallel update procedure converges to the unique NE of the game.

Corollary 1 says that stability under the Gauss-Seidel procedure coincides with stability under the Jacobi iteration. Furthermore, following the argument in [26], one can prove that any asynchronous computation where the players act at random times and use the most recently available policy from other players should converge to the NE, as long as no player remains idle for an infinite time duration. Hence, the achieved NE based on asynchronous updates coincides with the NE achieved with parallel or sequential updates.

The parallel and sequential PIWF algorithms are distributed. They both attempt to maximize the achievable sum-rate. Both have the same implementation complexity of the classic IWF algorithm. In Figure 5, we can see that the parallel PIWF converges faster than the sequential PIWF, especially for a large number of users. Whether the players act sequentially or in parallel makes a difference in the MAC design. We will discuss the impact of parallel PIWF and sequential PIWF on the MAC design in Section V. In this simulation, we assume that CRs are randomly located in a square area and 5 channels are available for their transmissions.
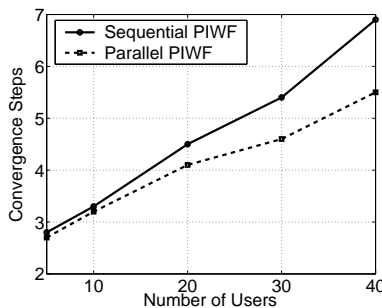
Fig. 5: Convergence to the NE solution in the sequential and parallel PIWF algorithms.

V. MAC PROTOCOL DESIGN

In this section, we describe a MAC protocol that allows CR users to operate efficiently in an opportunistic CRN. This protocol implements the distributed channel/power allocation strategies discussed in the previous sections. It should be noted that a number of multi-channel MAC protocols have been proposed in the context of CRNs (e.g., [14], [31], [30], and [27]). Most of them do not allow multiple CR transmissions within the same neighborhood to overlap in frequency channels, so there is no interference among CR users. Such a restriction simplifies the MAC design, but limits its spectrum efficiency. A natural extension (analogous to the improvement offered by the POWMAC protocol [17] over the classic CSMA/CA) is to allow CR users to overlap in spectrum, provided that their mutual interference does not lead to collisions. The IWF algorithm [29] and the no-regret algorithm [21] were proposed as two possible enabling techniques. However, the works in [29] and [21] provide only channel/power allocation algorithms and do not offer a practical MAC design.

In this section, we incorporate our price-based channel/power allocation algorithms into an operational MAC protocol. Since the IWF algorithm is a special case of the proposed PIWF algorithm, our MAC protocol can be simplified to accommodate the classic IWF algorithm, thus complementing the work in [29].

A. Assumptions

We consider a CRN with the following features:

- There exists a dedicated control channel or a coordinated control channel [30] that supports a community of CR users. Control packets are transmitted over the control channel using a pre-assigned power value P_{cont}.
- Channel gains between any two terminals are symmetric.

- The channel gain is static for the duration of several control packets and a flow of data packets.

B. Protocol Overview

Our MAC protocol uses the following three types of control packets for the handshaking between a CR transmitter and a CR receiver: Request-to-Send (RTS), Clear-to-Send (CTS), and Decide-to-Send (DTS). Unlike the classic CSMA/CA scheme and other multi-channel MAC protocols for CRNs, these control packets are not used to *exclusively* reserve channels (i.e., prevent neighboring CRs from accessing the reserved channels), but rather to exchange some information within the neighborhood. Such information is used by terminals to determine their transmission parameters.

The control packets are exchanged within a certain duration, referred to as the *contention window* (CW). A CW can be initiated asynchronously by any CR user that has packets to transmit and that is not aware of any active CWs in its neighborhood. Such a user is referred to as a *master user*. Other CR users that follow the schedule of an ongoing CW are called *slave users*. Note that the master/slave designation of a user is dynamic, i.e., it changes with traffic and mobility conditions. The objective of the CW is to allow several pairs of CR nodes to repeatedly negotiate their transmission channels and powers. As shown in Figure 6, the CW is divided into two parts. The first part, referred to as the *admission window* (AW), is used by CR nodes to compete for admission to the CW and initialize their transmission policies. The second part, referred to as the *training window* (TW), is used by the CR nodes to repeatedly negotiate their channel/power policies (as explained later). Note that the AW can be considered as the first iteration of the training process. CR nodes that have been successfully admitted during the CW transmit a flow of data packets over one or multiple data channels (as determined during the CW) within a *data window* (DW). The durations for the AW and DW are changed adaptively, similar to the single-channel POWMAC protocol [17]. As for TW, its size (in slots) is dictated by the convergence speed of the iterative resource allocation algorithm. In general, an unnecessarily large value increases the overhead, but does not necessarily improve the throughput (as shown in Figure 4). On the other hand, a small value may give sub-optimal results. In Section VI, we study the performance of the MAC protocol for various TW sizes.

C. Operation Details

1) Access Window: When a CR node A intends to establish a connection with another node B, it first needs to contend during the AW. If node A is not aware of any ongoing AW in its neighborhood, it
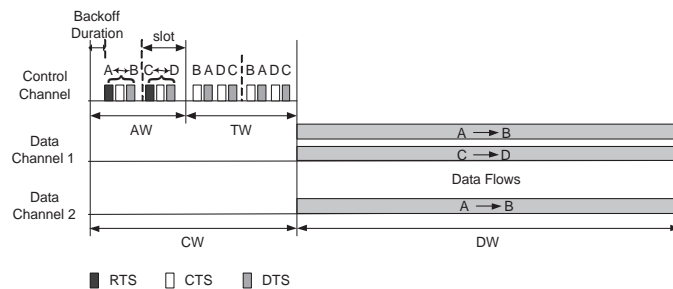
Fig. 6: Overview of the MAC operation with two CR transmissions ($A \to B$ and $C \to D$).

initiates a new AW (i.e., it becomes a master user). Otherwise, node A contends during one of the slots of the ongoing AW. In either case, node A first backs off by a random amount of time, selected from $[T_{min}, T_{max}]$, before accessing the channel.

The AW consists of a number of fixed-size slots. The size of each slot is T_{max} plus the durations of the RTS, CTS and DTS packets, plus 3 SIFS durations (SIFS denotes the short interframe spacing between successive control packets). In each slot, CR nodes compete for admission following a standard CSMA approach.

If CR B successfully receives the RTS packet from A, it needs to decide the initial channel/power policy for the link $A \to B$. This is done as follows:

- First, node B estimates the channel gain between itself and node A (denoted by $h_{AB}(0)$). This is facilitated by knowledge of the RTS's transmission power (P_{cont}) and the received power of the RTS. From $h_{AB}(0)$, CR B computes $h_{AB}(k)$ for all $k = 1, 2, \ldots, K$. The determination of $h_{AB}(k)$ from $h_{AB}(0)$ is made possible by knowing the carrier frequencies and by assuming a certain path-loss model. For example, under the two-ray model [23] and for a given transmission power, we have $h_{AB}(k) = h_{AB}(0) \times (f_0/f_k)^2$, where f_0 is the carrier frequency of the control channel.

- Next, node B measures the TNPI $\mathbf{M}_B$ over all data channels. Note that for the sequential PIWF algorithm, if there are previous CTS/DTS packets that have been received in the same AW, $\mathbf{M}_B$ is computed as the sum of the current $\mathbf{M}_B$ and the predicted CR-to-CR interference, which is obtained by assuming that the neighboring links transmit using the channels/powers specified in their CTS/DTS.

- Then, node B determines the pricing factor $\lambda_B(k)$ for all data channels k. For the sequential PIWF algorithm, $\lambda_B(k)$ is computed using (6), where the neighborhood information is obtained from previously received CTS/DTS packets in the same AW. For the parallel PIWF algorithm, $\lambda_B(k)$ is initialized to 0.

- Finally, based on the above information, node B decides its best-response transmission policy according to Proposition 3.

After the above procedures have been executed, node B will send a CTS, announcing its channel/power allocation. The CTS includes $M_B(k)$ and $h_{AB}(k)$ for all $k \in S_B$, which are used by neighboring CRs to update their best responses. Note that even if the set of selected channels S_B is empty (i.e., the computed transmission power is zero for all channels), the link $A \rightarrow B$ will still be admitted in the AW. This is because the data transmission $A \rightarrow B$ may later be allowed to proceed after several iterations in the TW.

If node A receives the CTS from B, it will respond with a DTS packet, repeating the information included in the CTS. This DTS is used to alleviate the *hidden terminal* problem as in [17]. The above procedures are repeated by CR pairs in every AW slot.

2) Training Window: CR nodes that are admitted in the AW iteratively negotiate their transmission policies in the TW, following the same order of their admission in the AW. In contrast to the AW, the TW is accessed in a TDMA manner. It consists of a number of slots (the TW size), where each slot is used to conduct one iteration of the channel/power allocation algorithm, using CTS and DTS packets. Note that there is no need for the RTS during the TW, since new admissions are not allowed.

In each iteration, the receiver of a CR link updates the transmission policies based on the policies of its neighbors. The updates are made based on either the sequential or the parallel scheme. Specifically, if the sequential PIWF algorithm is applied, the transmission policy of each CR user is made based on the policies of all previous users in the same iteration (obtained from CTS/DTS) and those of the other users in the previous iteration, as described in Algorithm 1. If the parallel PIWF algorithm is applied, the policy of each CR user is made based on the policies of other CR users in the previous iteration, as described in Algorithm 2. Note that the AW is regarded as the initial iteration of the training process. After each computation, the receiver sends a CTS, announcing its transmission policy. Upon receiving the CTS, the transmitter will send a DTS to repeat the information included in the CTS.

3) Data Window: The last negotiated transmission policies in the TW are used by the CR nodes for data transmissions in the DW. In the DW, a flow of data packets is transmitted from each CR transmitter. The length of the flow is selected such that the channel conditions remain static over the entire flow. Obviously, the DW size needs to be selected according to the channel's coherence time.

D. Simplified Packet-based MAC Design

The above MAC design can be used for flow-based channel access, where a flow of data packets is transmitted using converged channel/power policies agreed upon during the TW. Thus, the sum-rate of all competing CRs are likely to be maximized if the channels remain static over the entire data flow. However, if sum-rate optimality is not critical, we can simplify the protocol by removing the TW and only allow for single data-packet transmission in the DW. This design then becomes packet-based, and the convergence is now achieved after several sessions of CW and DW if channel conditions remain static within this period.

Note that in the previous section, all CR nodes contend in the AW with equal probability. In contrast, in the packet-based MAC design, the admitted users in the previous AW have priorities in accessing the control channel over other CR users. Specifically, the admitted links in the previous AW will contend in the current AW without backoff, according to their order in the previous AW, as long as they still have packets to transmit. After these links have been admitted, other links compete for the remaining slots, following the backoff mechanism that was discussed in the previous section. Such a design is meant to facilitate the convergence behavior.

The channel/power policies are updated in the AW following similar procedures to the flow-based MAC. The only difference is that the TNPI level is now estimated from the previous DW, instead of the previous iteration in the TW. In the next section, we compare the performance of this design with that of the flow-based MAC.

VI. PERFORMANCE EVALUATION

To evaluate the effectiveness of the proposed MAC, we conduct numerical experiments using CSIM; a process-oriented discrete-event programming environment that is based on the C language. We simulate a hybrid network with one PRN and one CRN. Nodes in these networks are uniformly distributed over a square area of length 100 meters. The PRN consists of 10 PR nodes that operate in the 300 MHz band, occupying five non-overlapping 1-MHz channels. The time is divided into slots, each of length 10 ms. In each slot, each PR attempts to transmit with a probability α (the PR's activity factor) and randomly selects a channel. The transmission power of each PR is 1 Watt, and the antenna length is 5 cm.

We simulate 10 pairs of CR nodes. The maximum transmission power for a CR is 1 Watt. The AWGN noise level is set to -70 dBm over all channels. The maximum transmission range of each CR is set to

45 meters, so the CRN has multiple neighborhoods with hidden terminals. Each CR transmitter generates flows according to a Poisson process with parameter λ flows/second. Each flow has an exponentially distributed duration of mean $1/\mu$ seconds. The traffic rate for a CR is defined as λ/μ. We set the CR-to-PR power mask to 0.5 Watt for all channels. The PR-to-CR and CR-to-CR interference are simulated using a two-ray model [23].

We study the performance of the flow-based and packet-based PIWF-MAC protocols, and contrast them with a flow-based IWF-MAC protocol. Since the IWF algorithm is a special case of the PIWF algorithm, our MAC protocols are also applicable to the classic IWF algorithm. We compare the performance in terms of the system throughput and the average power consumption.

Figure 7(a) shows the system throughput versus the traffic rate. As expected, the flow-based PIWF-MAC protocol gives the highest throughput. The throughput improvement over IWF-MAC becomes more significant with higher traffic rates. It is interesting to see that the simplified packet-based PIWF-MAC protocol exhibits comparable system throughput with the flow-based PIWF-MAC protocol. Besides achieving a higher throughput, the PIWF-MAC protocols also achieve a significant reduction in the transmission power, as shown in Figure 7(b). This is because in the classic IWF algorithm, users greedily maximize their own rates by sending at the maximum transmission power, while such a behavior is tampered by the pricing technique used in PIWF.

Figure 8(a) shows the throughput versus α. As expected, a higher α results in a higher PR-to-CR interference, which negatively affects the system throughput. Figure 8(b) shows the corresponding average power consumption. In all cases, the PIWF-MAC protocols consume less power than the IWF-MAC protocol.

Finally, Figure 9(a) shows the throughput versus the TW size. Since the simplified packet-based PIWF-MAC does not use a TW, we only compare the flow-based PIWF-MAC with the flow-based IWF-MAC. Intuitively, a larger TW size will ensure that CR users converge to the NE. However, as was seen in Figure 4, 2-3 iterations are normally sufficient to reach a near-optimal sum-rate. Figure 9(a) shows that setting the TW size to 2 is enough to reach 95% of the maximum throughput. Figure 9(b) shows the corresponding average power consumption.

VII. CONCLUSIONS

In this paper, we proposed a PIWF algorithm for spectrum sharing in cognitive radio networks. Our PIWF algorithm can be implemented distributively with CRs repeatedly negotiating their transmission
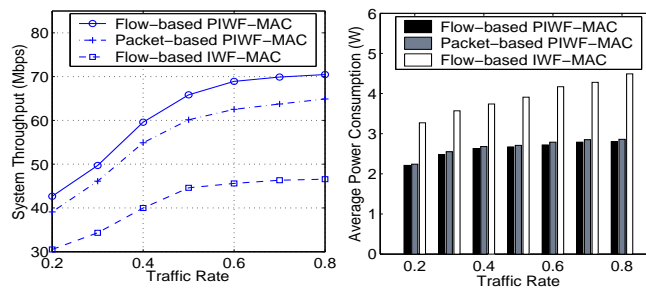
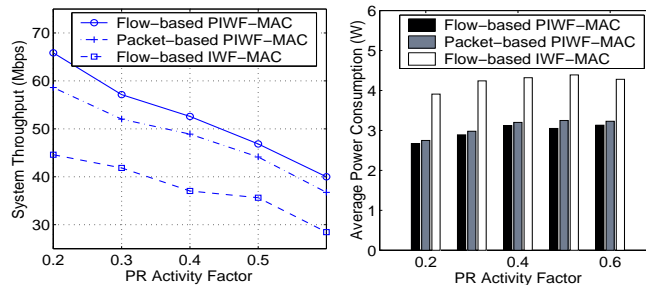Fig. 7: Performance under a fixed $\alpha = 0.2$.

Fig. 8: Performance under a fixed traffic rate $\lambda/\mu = 0.5$.

powers and spectrum. Simulation results showed that the proposed algorithm greatly improves the NE compared with the one achieved using the IWF approach. Based on the order by which CR nodes make their resource allocation decisions, we studied sequential and parallel versions of the PIWF algorithm. The parallel update scheme was shown to converge faster than the sequential update scheme, especially for a large number of users. Based on the PIWF algorithms, flow-based and packet-based MAC protocols were designed. Our simulation results showed that the PIWF-MAC protocol achieves considerably higher system throughput compared with the IWF-MAC, with less energy consumption.

REFERENCES

[1] P. Antoniadis, C. Courcoubetis, and R. Mason. Comparing economic incentives in peer-to-peer networks. *Computer Networks*, 46(1):133–146, 2004.

[2] T. Basar. Relaxation techniques and asynchronous algorithms for on-line computation of non-cooperative equilibria. *Journal of Economic Dynamics and Control*, 11(4):531–549, 1987.

[3] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[4] R. Cendrillon, W. Yu, M. Moonen, J. Verlinden, and T. Bostoen. Optimal multi-user spectrum management for digital subscriber lines. *IEEE Transactions on Communications*, 50(2):291–303, February 2006.

[5] Q. Chen and Z. Niu. A game-theoretical power and rate control for wireless ad hoc networks with step-up price. *IEICE Transactions on Communications*, E88-B(9):3515–3523, September 2005.
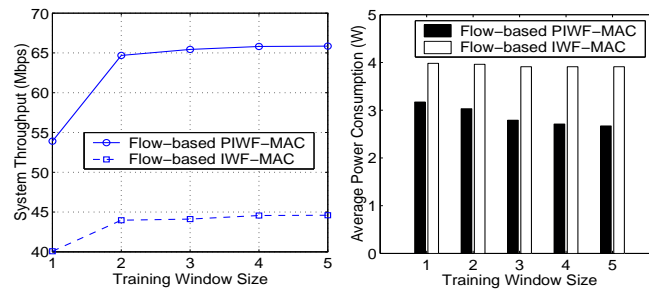
Fig. 9: Performance under traffic rate $\lambda/\mu = 0.5$ and $\alpha = 0.2$.

[6] S. T. Chung, S. J. Kim, J. Lee, and J. M. Cioffi. A game-theoretic approach to power allocation in frequency-selective Gaussian interference channels. In *Proceedings of the IEEE International Symposium on Information Theory (ISIT 2003)*, page 316, June 29-July 4, 2003.

[7] S. Cui, J. Xiao, A. J. Goldsmith, Z.-Q. Luo, and H. V. Poor. Estimation diversity and energy efficiency in distributed sensing. To appear in *IEEE Transactions on Signal Processing*, 2007.

[8] FCC. ET Docket No. 03-322, Notice of proposed rule making and order. December 2003.

[9] S. Haykin. Cognitive radio: Brain-empowered wireless communications. *IEEE Journal on Selected Areas in Communications*, 23(2):201–220, February 2005.

[10] J. Hirshleifer, A. Glazer, and D. Hirshleifer. *Price Theory and Applications Decisions, Markets, and Information*. Cambridge University Press, November 2005.

[11] Y. T. Hou, Y. Shi, and H. D. Sherali. Optimal spectrum sharing for multi-hop software defined radio networks. In *Proceedings of the IEEE INFOCOM Conference*, Anchorage, AL, May 2007.

[12] J. Huang, R. Cendrillon, M. Chiang, and M. Moonen. Autonomous spectrum balancing (ASB) for frequency selective interference channels. In *Proceedings of the IEEE International Symposium on Information Theory (ISIT 2006)*, July 2006.

[13] Z. Ji and K. J. R. Liu. Dynamic spectrum sharing: A game theoretical overview. *IEEE Communications Magazine*, 45(5):88–94, May 2007.

[14] S. Krishnamurthy, M. Thoppian, S. Venkatesan, and R. Prakash. Control channel based MAC-layer configuration, routing and situation awareness for cognitive radio networks. In *Proceedings of the Military Communications Conference (MILCOM)*, volume 1, pages 455 – 460, Oct 2005.

[15] R. Maheswaran and T. Basar. Decentralized network resource allocation as a repeated noncooperative market game. In *Proceedings of the 40th IEEE Conference on Decision and Control*, volume 5, pages 4565–4570, Orlando, FL, Dec 2001.

[16] J. Mitola. *Cognitive radio: An integrated agent architecture for software defined radio*. Ph.D. Dissertation, KTH Royal Institute of Technology, Stockholm, Sweden, 2000.

[17] A. Muqattash and M. Krunz. POWMAC: A single-channel power-control protocol for throughput enhancement in wireless ad hoc networks. *IEEE Journal on Selected Areas in Communications*, 23(5):1067–1084, May 2005.

[18] J. Neel, J. Reed, and R. Gills. The role of game theory in the analysis of software radio networks. In *Proceedings of the SDR Forum Technical Conference*, November 2002.

[19] J. Neel, J. Reed, and R. Gills. Convergence of cognitive radio networks. In *Proceedings of the Wireless Communications and Networking Conference*, March 2004.

[20] M. J. Neely. Optimal pricing in a free market wireless networks. In *Proceedings of the IEEE INFOCOM Conference*, Anchorage, AL,

May 2007.

[21] N. Nie and C. Comaniciu. Adaptive channel allocation spectrum etiquette for cognitive radio networks. In *IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks (IEEE DySPAN)*, November 2005.

[22] M. J. Osborne. *An Introduction to Game Theory*. Oxford University Press, 2004.

[23] T. S. Rappaport. *Wireless Communications - Principles and Practice*. Prentice Hall, 2nd edition, 2001.

[24] J. B. Rosen. Existence and uniqueness of equilibrium points for concave N-person games. *Econometrica*, 33(3):520–534, July 1965.

[25] C. U. Saraydar, N. B. Mandayam, and D. J. Goodman. Efficient power control via pricing in wireless data networks. *IEEE Transactions on Communications*, 50(2):291–303, February 2002.

[26] G. Scutari, D. P. Palomar, and S. Barbarossa. Asynchronous iterative waterfilling for Gaussian frequency-selective interference channels: A unified framework. Submitted to *IEEE Transactions on Information Theory*, August 2006.

[27] T. Shu, S. Cui, and M. Krunz. Medium access control for multi-channel parallel transmission in cognitive radio networks. In *Proceedings of the IEEE GLOBECOM Conference*, November 2006.

[28] F. Wang, O. Younis, and M. Krunz. Throughput-oriented MAC for mobile ad hoc networks: A game-theoretic approach. submitted to *ACM/Kluwer Wireless Networks (WINET) journal*, 2006.

[29] W. Yu. *Competition and cooperation in multi-user communication environments*. Ph.D. Dissertation, Stanford University, Stanford, CA, 2002.

[30] J. Zhao, H. Zheng, and G. Yang. Distributed coordination in dynamic spectrum allocation networks. In *IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks (IEEE DySPAN)*, November 2005.

[31] Q. Zhao, L. Tong, A. Swami, and Y. Chen. Decentralized cognitive MAC for opportunistic spectrum access in ad hoc networks: A POMDP framework. *IEEE Journal on Selected Areas in Communications*, 25(3):589–600, April 2007.