Quality of Service Games for Spectrum Sharing

Richard Southwell, Xu Chen, Member, IEEE, and Jianwei Huang, Senior Member, IEEE

Abstract—Today’s wireless networks are increasingly crowded with an explosion of wireless users, who have greater and more diverse quality of service (QoS) demands than ever before. However, the amount of spectrum that can be used to satisfy these demands remains finite. This leads to a great challenge for wireless users to effectively share the spectrum to achieve their QoS requirements. This paper presents a game theoretic model for spectrum sharing, where users seek to satisfy their QoS demands in a distributed fashion. Our spectrum sharing model is quite general, because we allow different wireless channels to provide different QoS, depending upon their channel conditions and how many users are trying to access them. Also, users can be highly heterogeneous, with different QoS demands, depending upon their activities, hardware capabilities, and technology choices. Under such a general setting, we show that it is NP hard to find a spectrum allocation which satisfies the maximum number of users’ QoS requirements in a centralized fashion. We also show that allowing users to self-organize through distributed channel selections is a viable alternative to the centralized optimization, because better response updating is guaranteed to reach a pure Nash equilibrium in polynomial time. By bounding the price of anarchy, we demonstrate that the worst case pure Nash equilibrium can be close to optimal, when users and channels are not very heterogeneous. We also extend our model by considering the frequency spatial reuse, and consider the distributed QoS demand satisfaction problem among the users. Game theory is a useful tool for designing distributed algorithms that allow users to self-organize, optimize their QoS requirements. This approach puts most of the implementation complexity at the operator side, and wireless devices do not need to be very sophisticated. However, as the networks grow larger and more heterogeneous, this approach can become unsuitable for two reasons. Firstly, the QoS demands of wireless users are highly heterogeneous, which implies that the operator needs to gather massive amounts of information from users in order to perform the centralized optimization. Secondly, finding the system-wide optimal QoS demand satisfaction solution is computationally challenging – in fact we show that it is NP hard. It is hence difficult for the operator to compute the optimal solution to meet users’ real-time QoS demands. The alternative approach is a decentralized approach, where each wireless user makes the spectrum access decision locally to meet its own QoS demand, while taking the network dynamics and other users’ actions into consideration. This is feasible since new technologies like cognitive radio give users the ability to scan and switch channels easily. The decentralized approach enables more flexible spectrum sharing, scales well with the network size, and is particular suitable when users belong to multiple network entities.

In this paper, we focus on the decentralized approach, and propose a new framework of QoS satisfaction games to model the distributed QoS demand satisfaction problem among the users. Game theory is a useful tool for designing distributed algorithms that allow users to self-organize, optimize their channel selections, and satisfy their QoS demands. Our QoS satisfaction game framework is developed, based on the theory of congestion games [2]. The central idea behind congestion games is that there are many players, each of which selects a resource to use. A player’s utility is a non-increasing function

Fig. 1. A Venn diagram showing our results on different types of QoS satisfaction games.
of the total number of players using the same resource. The distributed QoS satisfaction problem can be modeled using congestion games by thinking of the players as wireless users, while the resources represent different channels [3]. The satisfaction of a user’s QoS demand depends on its congestion level, i.e., how many users are competing for its channel. In our QoS satisfaction game, a player achieves a unit utility when its channel’s data rate is sufficiently high to satisfy its QoS demand. Otherwise, the player’s utility is negative and it is better off by switching channels (to improve the payoff) or turning off its transmitter (to receive a zero payoff).

A. Related Work

Rosenthal proposed the original congestion game model [2] for the scenario where different resources can have different utility functions associated with them (i.e., heterogenous resources) but all players have the same utility function for any particular resource (i.e., homogenous players). This kind of system has a pleasing feature known as the finite improvement property - which means that when the system evolves because players asynchronously perform better response updates (i.e., the players selfishly improve their resource choices), the system is guaranteed to reach a pure Nash equilibrium in a finite number of steps. A pure Nash equilibrium is a system state where no player has any incentive to deviate unilaterally.

However, the original congestion game is not general enough to model spectrum sharing, because it assumes that players are homogenous, whereas wireless network users are often highly heterogenous. The congestion games with player-specific utility functions considered in [4] are more appropriate for this modeling purpose. Authors in [3], [5]–[7] have adopted such a game model for studying spectrum sharing problems. However, unlike classical congestion games, these games are not necessarily guaranteed to possess the finite improvement property.

Spatial reuse is another feature of wireless networks that the original congestion game model does not account for. In reality only nearby users on the same channel will interfere with each other. Users which are distantly separated will not cause congestion to each other. A congestion game on a graph can be used to realistically capture the spatial aspect of spectrum sharing. The idea behind such a system is that a user’s utility only depends upon the number of users of the same channel who are linked to them in the graph. In [8], we introduced a general class of congestion games on graphs that are appropriate for modeling spectrum sharing. Although there are many subclasses of these games which always admit the finite improvement property, we demonstrated that there exist congestion games on graphs that do not have any pure Nash equilibria. We have also further developed several more elaborate graphical congestion game models [9]–[12] with applications to spectrum sharing.

A common assumption within most previous congestion game based spectrum sharing literature (e.g., [3], [5]–[7], [9], [12]) is that a user’s utility strictly increases with its received data rate (and hence strictly decreases with the congestion level). This is true, for example, when users are running elastic applications such as file downloading. However, there are many other types of applications with more specific QoS requirements, such as VoIP and video streaming. These inelastic applications cannot work properly when their QoS requirements (e.g., data rates) are unmet, and do not enjoy any additional benefits when given more resources than needed. This kind of traffic is becoming increasingly popular over the wireless networks (e.g., mobile video traffic exceeded 50% percent of all wireless traffic in 2011 according to the report by Cisco [13]). This motivates the QoS satisfaction game model in this paper.

Rather than assuming that users wish to increase their data rates whenever possible, we assume that each user has a fixed QoS demand. If the demand is satisfied, then the user has no inclination to change his choice of resource. Our game model was inspired by the games in satisfaction form considered in [14]. In [14] the authors considered other games where players wish to satisfy demands, and the authors design algorithms to find satisfaction equilibria, which are strategy profiles where all users are satisfied. In our paper, we consider the more general case where some users’ QoS requirements may not be satisfied (given the limited spectrum resource). The case where a satisfaction equilibrium exists becomes a special case of our model. Moreover, we also take into account the issue of spatial reuse. This makes the modeling more practical for wireless communication systems. The generalization considered in our model result in more challenges and significant differences in analysis.

When discussing the achievability of the equilibrium, we focus on dynamics where one player can perform a better response update each time. There are many alternative types of dynamics we could consider, such as smoothed best response dynamics and imitation dynamics [15]. We could also consider the replicator dynamics from evolutionary game theory. Reference [16] showed that replicator dynamics can be used for spectrum sharing using appropriate message passing protocols. Replicator dynamics is most useful when the user population is large, and in which case the system will follow continuous (essentially deterministic) dynamics which normally converge to evolutionarily stable strategies. However, many techniques from evolutionary game theory rely upon the assumption the players are homogenous, while our wireless users are typically heterogenous. Another issue is that translating replicator dynamics into the spatial setting (i.e., a game on a graph) is quite difficult.

B. Contributions

Our main results and contributions can be summarized as follows (and are illustrated in Figure 1):

- A general QoS satisfaction game framework: We formulate the distributed QoS demand satisfaction problem among wireless users as a QoS satisfaction game, which is general enough to capture the details of spectrum sharing over a wide range of scenarios, with heterogenous channels and users. Despite allowing for heterogenous channels, heterogenous users, and spatial interactions, we still obtain several significant analytic results.
- Remarkable convergence properties: We prove that every QoS satisfaction game has the finite improvement property. This is remarkable because many congestion games.
with heterogenous resources and players do not have this feature. More importantly, it enables us to design a distributed QoS satisfaction algorithm which allows wireless users to easily self-organize into a pure Nash equilibrium.

- Spatial generalization: We generalize the model by thinking of users as vertices, which are linked in a graph and can only interfere with their neighbors. We show that the resulting QoS games on graphs also possess the finite improvement property.

The rest of the paper is organized as follows. We introduce the QoS satisfaction game model and study its properties in Sections II and III, respectively. We then generalize the game model with spatial reuse in Section IV. We then propose the distributed QoS satisfaction algorithm and evaluate its performance by simulations in Section V. Finally, we conclude in Section VI. Most proofs are provided in the online technical report in [17].

II. QoS Satisfaction Game

In this section we formally define the QoS satisfaction game model for spectrum sharing. Spectrum sharing is a promising approach to address the spectrum under-utilization problem. Field measurements by Shared Spectrum Cooperation in Chicago area shows that the overall average utilization of a wide range of different types of spectrum bands is lower than 20% [18]. In order to improve the overall spectrum utilization, several countries have recently reformed their policy (such as the FCC’s ruling for the TV white space [19]) and allow spectrum sharing, such that unlicensed users equipped with cognitive radios can access the channels which are tentatively not used by the licensed spectrum users. In this paper, we consider the spectrum sharing problem among multiple unlicensed users who run different applications and hence have heterogeneous QoS demands.

A. Game model

A QoS satisfaction game is defined by a tuple 
\((\mathcal{N}, \mathcal{C}, (Q_n^c)_{n \in \mathcal{N}, c \in \mathcal{C}}, (D_n)_{n \in \mathcal{N}})\) where:

- \(\mathcal{N} = \{1, \ldots, N\}\) is the set of wireless unlicensed users, also referred as the players.
- \(\mathcal{C} = \{1, \ldots, C\}\) is the set of channels. Each unlicensed user may select one channel to access. Furthermore, we introduce the element 0 to represent the dormant state. Choosing the dormant state will be beneficial when an unlicensed user’s QoS demand cannot be satisfied due to limited resources. In such a case the user can choose the dormant state 0, which corresponds to ceasing its transmission to save power consumption. Now we use the term ‘dormant state’ instead of ‘virtual channel’ since it involves introducing less new concepts, and we no longer have to speak of “real” channels. In summary, each unlicensed user/player has a strategy set \(\tilde{\mathcal{C}} = \{0, 1, \ldots, C\}\) which consists of all channels, together with the dormant state. The strategy profile of the game is given as \(\mathbf{x} = (x_1, x_2, \ldots, x_N) \in \tilde{\mathcal{C}}^N\), where each unlicensed user \(n\) chooses a strategy \(x_n \in \tilde{\mathcal{C}}\).

- \(Q_n^c(\cdot)\) is a non-increasing function that characterizes the data rate received by an unlicensed user \(n\) who has selected channel \(c\). Specifically, we have \(Q_n^c(I^c(x)) = \theta_n B_c g_n^c(I^c(x))\), with \(I^c(x) = |\{n \in \mathcal{N} : x_n = c\}\) being the congestion level of channel \(c\), i.e., the number of users who choose channel \(c\). We detail the parameters in \(Q_n^c(\cdot)\) as follows.

1. \(\theta_n^c \in \{0, 1\}\) is the channel availability indicator. When channel \(c\) is occupied by licensed users and not available for unlicensed user \(n\), we have \(\theta_n^c = 0\), in which case \(Q_n^c(I^c(x)) = 0\) for any value of \(I^c(x)\). For a limited period of time, the usage of spectrum by licensed users is assumed to be static (but can change in different periods)\(^1\). This is appropriate for modeling the TV spectrum, for example, where the activities of licensed users change very slowly. According to the most recent ruling by the FCC, unlicensed users can reasonably and accurately determine the spectrum availability within a short amount of time by consulting a database [19]. When channel \(c\) is available for the spectrum access by an unlicensed user \(n\) (i.e., \(\theta_n^c = 1\)), we have \(Q_n^c(I^c(x)) > 0\).

2. \(B_c^n\) is the mean channel throughput of user \(n\) on channel \(c\). We allow user specific throughput functions, i.e., different users may have different \(B_c^n\) even on the same channel \(c\). This enables us to model users with different transmission technologies, different coding/modulation schemes, different channel conditions, and different reactions to the same licensed user on the channel. For example, we can compute the maximum channel throughput \(B_c^n\) according to the Shannon capacity as

\[
B_c^n = W_c \log_2 \left(1 + \frac{\zeta_n z_n^c}{\omega_n^c}\right),
\]

where \(W_c\) is the bandwidth of channel \(c\), \(\zeta_n\) is the fixed transmission power adopted by user \(n\) according to the requirements such as the primary user protection, \(\omega_n^c\) denotes the background noise power, and \(z_n^c\) is the user-specific channel gain.

3. \(g_n^c(I^c(x))\) is the channel contention function that describes the probability that user \(n\) can successfully grab the channel \(c\) for data transmissions given the congestion level \(I^c(x)\). In general, \(g_n^c(I^c(x))\) decreases as the number of contending users \(I^c(x)\) increases. For example, if we adopt the TDMA mechanism for the medium access control (MAC) to schedule users in the round-robin manner, then we have \(g_n^c(I^c(x)) = \frac{1}{I^c(x)}\).

- \(D_n \geq 0\) is the data rate demand of unlicensed user \(n\). For example, listening to an MP3 online will require a small \(D_n\), whereas watching a high definition streaming

\(^{1}\)We show in Theorem 7 that the proposed QoS satisfaction game algorithm can converge in a fast manner (e.g., less than one second in practical 802.11 systems). In this case, as long as the activities of licensed users change in a larger timescale in terms of seconds/minutes/hours (e.g., TV/daytime radio broadcasting), we can still implement the QoS satisfaction game solution for the system.
video requires a large $D_n$.

The utility of an unlicensed user $n$ in strategy profile $\mathbf{x}$ is

$$U_n(\mathbf{x}) = \begin{cases} 1, & \text{if } x_n \neq 0 \text{ and } Q^x_n(I^x_n(\mathbf{x})) \geq D_n, \\ 0, & \text{if } x_n = 0, \\ -1, & \text{if } x_n \neq 0 \text{ and } Q^x_n(I^x_n(\mathbf{x})) < D_n. \end{cases} \quad (2)$$

A satisfied user is an unlicensed user $n$ who chooses a channel $x_n \neq 0$ and receives a data rate $Q^x_n(I^x_n(\mathbf{x}))$ not smaller than its QoS demand $D_n$. A satisfied user receives a utility of 1. A dormant user is an unlicensed user $n$ choosing the dormant state $x_n = 0$. Such a dormant user does not receive any benefit (as it achieves a zero data rate) or any penalty (as it does not waste any energy), and gets a utility of $U_n(\mathbf{x}) = 0$. A suffering user is an unlicensed user $n$ who chooses a channel $x_n \neq 0$ but receives a data rate $Q^x_n(I^x_n(\mathbf{x}))$ below its QoS demand $D_n$. Such a suffering user expends power without gaining any benefit, and so it gets a utility of $U_n(\mathbf{x}) = -1$.

A suffering user can always increase their utility by becoming dormant without harming any other user. This suggests that rational (i.e., utility maximizing) players will eventually end up at strategy profiles which contain no suffering users. We say that a strategy profile is natural if it holds no suffering users.

It is worth noting that we can easily generalize our model by allowing an unlicensed user $n$ to receive a utility of $u_n$ if it is satisfied, $v_n$ if it is dormant, and $t_n$ if it is suffering, where $u_n > v_n > t_n$. Making this generalization does not affect the better response dynamics or the set of pure Nash equilibria discussed later on, because the preference orderings of the strategies in the generalized game are the same as in our current model\(^2\). Our results about convergence (Theorem 1) and computational complexity (Theorem 2) also remain true for games with generalized utility functions. However, since the generalized games allow different players to receive different utilities when satisfied, our results about social optimality (Theorems 4 and 5) may not hold for the generalized games. In this paper, we will restrict our attention to the utility choices when satisfied user $A_n$ is an unlicensed user

$$\begin{align*}
\text{Social Welfare} & : \sum_{n=1}^{N} U_n(\mathbf{x}) \text{ of a strategy profile } \mathbf{x} \text{ is the sum of all players’ utilities.} \\
\text{Social Optimum} & : \text{A strategy profile } \mathbf{x} \text{ is a social optimum when it maximizes social welfare.} \\
\text{Better Response Update} & : \text{The event where a player } n \text{ changes its choice of strategy from } x_n \text{ to } c \text{ is a better response update if and only if } U_n(c, \mathbf{x}_{-n}) > U_n(x_n, \mathbf{x}_{-n}), \text{ where we write the argument of the function as } \mathbf{x} = (x_n, \mathbf{x}_{-n}) \text{ with } \mathbf{x}_{-n} = (x_1, \ldots, x_{n-1}, x_{n+1}, \ldots, x_N) \text{ representing the strategy profile of all players except player } n. \\
\text{Definition 4 (Pure Nash Equilibrium).} & \text{A strategy profile } \mathbf{x} \text{ is a pure Nash equilibrium if no players at } \mathbf{x} \text{ can perform a better response update, i.e., } U_n(x_n, \mathbf{x}_{-n}) \geq U_n(c, \mathbf{x}_{-n}) \text{ for all } c \in C \text{ and } n \in N. \\
\text{Definition 5 (Finite Improvement Property).} & \text{A game has the finite improvement property if any asynchronous better response update process}^3 \text{ terminates at a pure Nash equilibrium within a finite number of updates.}
\end{align*}$$

C. Transformation to an equivalent interference threshold form

For the discussion convenience, we will introduce an equivalent interference-threshold form of the QoS satisfaction game. The key idea is to relate a user’s received congestion level with its QoS demand satisfaction. Since the data rate function $Q^c(I^c)$ is non-increasing with the congestion level $I^c$, there must exist a critical threshold value $T^c_n$ such that $Q^c(I^c) \geq D_n$ if and only if the congestion level $I^c \leq T^c_n$ (see Figure 2 for an illustration). Formally, given a pair of $(Q^c_n, D_n)$, we shall define the threshold $T^c_n$ of channel $c$ with respect to user $n$ to be an integer such that

$$\begin{align*}
\text{If } Q^c_n(I^c) < D_n \text{ for each } I^c \in N, \text{ then } T^c_n = 0 \text{ (hence user } n\text{'s QoS demand can never be satisfied on channel } c \text{ even if it is the only user on this channel),} \\
\text{If } Q^c_n(I^c) > D_n \text{ for each } I^c \in N, \text{ then } T^c_n = N + 1 \text{ (hence user } n\text{'s QoS demand is always satisfied on channel } c \text{ even if all users use this channel)} \quad (3) \\
\text{Otherwise } T^c_n \text{ is equal to the maximum integer } I^c \in N \text{ such that } Q^c_n(I^c) \geq D_n.
\end{align*}$$

These conditions guarantee that

$$Q^c_n(I^c) \geq D_n \iff I^c \leq T^c_n.$$ 

We can then express a QoS satisfaction game $g = (N, C, (Q^c_n)_{n \in N, c \in C}, (D_n)_{n \in N})$ in the interference threshold form $g' = (N, C, (T^c_n)_{n \in N, c \in C})$. And the utility of user $n$ can

\(^2\)Technically speaking, our game is weakly isomorphic [20] to this generalized version.

\(^3\)Where no more than one player updates his strategy at any given time.

\(^4\)It is possible to set $T^c_n = N + 1$ to bound the differences between thresholds, which helps the proof of fast convergence of the distributed algorithm in Theorem 7.
be computed accordingly as
\[ U_n(x) = \begin{cases} 
1, & \text{if } x_n \neq 0 \text{ and } I^x_n(x) \leq T_n^c, \\
0, & \text{if } x_n = 0, \\
-1, & \text{if } x_n \neq 0 \text{ and } I^x_n(x) > T_n^c. 
\end{cases} \] 

(4)

The interference threshold transformation reduces the size of parameters by replacing \((Q_n^c, D_n)\) with \(T_n^c\). Moreover, the result in (3) ensures that the original game \(g\) is equivalent to the game \(g'\), since the utility \(U_n(x)\) received by player \(n\) in \(g\) is the same as that received by player \(n\) in \(g'\) for every strategy profile \(x\) and player \(n\). For the rest of the paper, we will analyze the QoS satisfaction game in the interference threshold form. Note that Equations (2) and (4) are equivalent. It is just that we write the latter expression in terms of thresholds.

III. PROPERTIES OF THE QoS SATISFACTION GAME

Now we explore the properties of QoS satisfaction games, including the existence of pure Nash equilibria and the finite improvement property. We shall also describe the conditions including the existence of pure Nash equilibria and the finite improvement property. Moreover, any asynchronous better response update process is guaranteed to reach a pure Nash equilibrium. Theorem 1 is a direct consequence of the more general Theorem 7 in Section IV. Theorem 1 is very important, because it implies that the general QoS satisfaction games have a fast convergent property. Although Theorem 1 shows that pure Nash equilibria can be found relatively easily, it does not offer any insight into how to select the most beneficial pure Nash equilibria. Equilibrium selection seems to be a difficult problem in the general case. However, we show how to find pure Nash equilibria which are social optimum for special cases in Subsections III-D and III-E.

Theorem 2. The problem of finding a social optimum of a QoS satisfaction game is NP hard.

The problem of finding a social optimum of a QoS satisfaction game has some resemblance to the Knapsack problem (where items have different weights and values, and the objective is to maximize the value of items chosen without exceeding a given total weight threshold). The key difference is that the thresholds in our problem are associated with the players/items we are choosing, and there are multiple channels/knapsacks to allocate our players to. Our proof is based upon showing that the 3-dimensional matching decision problem (which is well known to be NP complete [21]) can be reduced to the problem of finding a social optimum of a QoS satisfaction game where thresholds \(T_n^c \in \{1, 3\}\) for each \(n\) and \(c\). Theorem 2 provides the major motivation for our game theoretic study, because it suggests that the centralized spectrum sharing problem is fundamentally difficult. It therefore makes sense to explore decentralized alternatives such as a game based spectrum sharing.

C. Price of Anarchy

Although Theorem 2 suggests that finding an optimal strategy profile can be very difficult, we do know from Theorem 1 that pure Nash equilibria can be found with relative ease. This naturally raises the question of how the social welfare of pure Nash equilibria compare to the maximum possible social welfare. In other words, how much social welfare can be lost by allowing the players to organize themselves, rather than being directed to a social optimum?

To gain insight into this issue, we study the price of anarchy [22]. Recall that \(\bar{c}^N\) is the set of strategy profiles of our game. Let \(\Xi \subseteq \bar{c}^N\) denote the set of pure Nash equilibria of our game. Note that Theorem 1 implies that \(\Xi\) is non-empty. The price of anarchy

\[ \text{PoA} = \frac{\max\left\{\sum_{n=1}^{N} U_n(x) : x \in \bar{c}^N\right\}}{\min\left\{\sum_{n=1}^{N} U_n(x) : x \in \Xi\right\}}, \] 

(5)

is defined to be the maximum social welfare of a strategy profile, divided by the minimum welfare of a pure Nash equilibrium. The social welfare of a system at a pure Nash equilibrium can be increased by at most PoA times by switching to a centralized solution.

Theorem 3. Consider a QoS satisfaction game \((N, C, (T_n^c)_{n \in N, c \in C})\), where \(T_n^c \geq 1\) for each player \(n\) and each channel \(c\). The PoA of this game satisfies

\[ \text{PoA} \leq \min \left\{ N, \max\left\{T_n^c : n \in N, c \in C\right\} \right\}, \] 

(6)

The constraint \(T_n^c \geq 1\) insures that some player will be satisfied in every pure Nash equilibrium of the game, and avoid the possibility of the PoA involving “division by zero”. Theorem 3 implies that the performance of every pure Nash equilibrium will be close to optimal when the minimum threshold of a user-channel pair is close to the maximum threshold of a user-channel pair. This is a very significant result, when one considers that pure Nash equilibria can be easily reached by better response updates (Theorem 1) while
finding social optima is NP hard (Theorem 2). Motivated by Theorem 3, we next study two special cases of QoS satisfaction games with homogenous settings, i.e., homogenous users and homogenous channels. In both cases, the social optimum can be actually achieved at a pure Nash equilibrium.

D. QoS satisfaction games with homogenous users

We first study the case of homogenous users. We say that a QoS satisfaction game has homogenous users when $T^c_n = T^c_n$ for each $c \in C$ (i.e., each player has the same threshold for any channel $c$). This corresponds to the case that all users have the same data rate function $Q^c_n$ on the same channel $c$ (but they may have different data rates on different channels) and the same demand $D_n$. For example, spectrum sharing in a network of RFID tags in a warehouse may correspond to such a QoS satisfaction game, because every device experiences the same environment and requires a similar data rate to operate.

When discussing QoS satisfaction games with homogenous users, we drop the superscripts and use $T_n$ to denote the common threshold of player $n$ for all channels. In Figure 4, we visualize the dynamics of QoS satisfaction games with homogenous channels by using arrows to represent potential better response updates. The update process can reach a pure Nash equilibrium according to Theorem 1.

Next we discuss the optimality of pure Nash equilibria. Firstly note that, a pure Nash equilibrium may not be a social optimum. For example, let us consider a game of six users, with thresholds $T_1 = T_2 = 2$, $T_3 = T_4 = 3$, and two channels. The game has a pure Nash equilibrium $x = (0, 0, 1, 1, 2, 2)$ with four satisfied users, which is not a social optimum. The strategy profile $y = (1, 2, 2, 2, 2)$, where all six users are satisfied, is a social optimum.

Second, a social optimum may not be a pure Nash equilibrium. We take the game with six users and thresholds $T_1 = 2$, $T_3 = T_4 = T_5 = 3$, and two channels as an example. The game has a social optimum $x = (1, 1, 2, 2, 2, 0)$ (with five satisfied users), which is not a pure Nash equilibrium because user 6 can do a better response update by switching to channel 1.

Surprisingly, there always exists a pure Nash equilibrium that is a social optimum for a game with homogenous channels. Moreover, we present an algorithm (Algorithm 1) that always generates a social optimum which is a pure Nash equilibrium. The key idea of the algorithm is to prioritize channel allocation according to users’ thresholds (i.e., the more severe congestion a user can tolerate, the higher priority it will get in channel allocation).

Algorithm 1 is a centralized algorithm that demonstrates the existence of a pure Nash equilibrium which is a social optimum. The distributed algorithm that globally converges to a Nash equilibrium (not necessarily socially optimal) will be discussed in Subsection V-A. Algorithm 1 begins by making

2) There are no suffering users in $x$ and the number of satisfied users is $\min\{N, \sum_{c=1}^C T^c\}$.

3) $x$ is a social optimum.

Theorems 1 and 4 together imply that any sufficiently long asynchronous better response updating sequence will converge to a social optimum in polynomial time when the game has homogenous users. Moreover, Theorem 4 implies that when $\sum_{c=1}^C T^c \geq N$, there exists a satisfaction equilibrium [14] where all the players can be satisfied.

E. QoS satisfaction games with homogenous channels

We next consider the case that the channels are homogenous. We say a QoS satisfaction game has homogenous channels when $T^1_n = T^2_n = \ldots = T^C_n$, for each user $n$ (i.e., all channels have the same threshold from any player’s perspective). This corresponds to the case that each user $n$ has the same data rate function $Q^c_n$ on all the channels, but different users may have different demands $D_n$. QoS satisfaction games with homogenous channels are highly relevant, because technologies such as frequency interleaving can be adopted in many wireless systems such as IEEE 802.11g networks [23] to make channels homogenous (i.e., having the same bandwidth and experiencing frequency flat fading).

When discussing QoS satisfaction games with homogenous channels, we drop the superscripts and use $T_n$ to denote the common threshold of player $n$ for all channels. In Figure 4, we visualize the dynamics of QoS satisfaction games with homogenous channels by using arrows to represent potential better response updates. The update process can reach a pure Nash equilibrium according to Theorem 1.

Next we discuss the optimality of pure Nash equilibria. Firstly note that, a pure Nash equilibrium may not be a social optimum. For example, let us consider a game of six users, with thresholds $T_1 = T_2 = 2$, $T_3 = T_4 = 3$, and two channels. The game has a pure Nash equilibrium $x = (0, 0, 1, 1, 2, 2)$ with four satisfied users, which is not a social optimum. The strategy profile $y = (1, 2, 2, 2, 2)$, where all six users are satisfied, is a social optimum.

Second, a social optimum may not be a pure Nash equilibrium. We take the game with six users and thresholds $T_1 = T_2 = 2$, $T_3 = T_4 = T_5 = T_6 = 4$, and two channels as an example. The game has a social optimum $x = (1, 1, 2, 2, 2, 2)$ (with five satisfied users), which is not a pure Nash equilibrium because user 6 can do a better response update by switching to channel 1.

Surprisingly, there always exists a pure Nash equilibrium that is a social optimum for a game with homogenous channels. Moreover, we present an algorithm (Algorithm 1) that always generates a social optimum which is a pure Nash equilibrium. The key idea of the algorithm is to prioritize channel allocation according to users’ thresholds (i.e., the more severe congestion a user can tolerate, the higher priority it will get in channel allocation).

Algorithm 1 is a centralized algorithm that demonstrates the existence of a pure Nash equilibrium which is a social optimum. The distributed algorithm that globally converges to a Nash equilibrium (not necessarily socially optimal) will be discussed in Subsection V-A. Algorithm 1 begins by making
Algorithm 1: Finds a pure Nash equilibrium that is a social optimum for a game with homogenous channels.

**Input:** A QoS satisfaction game with $C$ homogenous channels and $N$ players, who have thresholds $T_1 \geq T_2 \geq \ldots \geq T_N$.

**Output:** A social optimum which is a pure Nash equilibrium.

1. Let $x^0 = (x_1^0, x_2^0, \ldots, x_N^0) = (0, 0, \ldots, 0)$
2. for $n = 1$ to $N$ do
3.   if $\exists c \in C : I^c(x^{n-1}) < T_n$ then
4.     Let $c^* = \min\{c \in C : I^c(x^{n-1}) < T_n\}$
5.     Let $x^n = (x_1^{n-1}, x_2^{n-1}, \ldots, x_{n-1}^{n-1}, c^*, x_{n+1}^{n-1}, \ldots, x_N^{n-1})$
6.   else
7.     Let $x^n = x^{n-1}$
8. return $x^N$

Algorithm 1 is guaranteed to generate a pure Nash equilibrium that is also a social optimum.

**Theorem 5.** Algorithm 1 has a complexity of $O(CN^2)$ and generates a strategy profile that is both a social optimum and a pure Nash equilibrium of a QoS satisfaction game with $C$ homogeneous channels and $N$ users.

Next Theorem 6 gives a sufficient condition for the existence of a strategy profile where all players are satisfied, in a QoS satisfaction game with homogeneous channels.

**Theorem 6.** If $T_n \geq \lceil \frac{N}{C} \rceil$ holds for every user $n$ in the QoS satisfaction game with $C$ homogeneous channels and $N$ users, then there is a strategy profile $x$ within which every user is satisfied (which is a pure Nash equilibrium).

IV. Spatial QoS Satisfaction Game

In all the games considered so far, we have assumed that every pair of users are close enough to cause congestion to each other, when they use the same channel. However, in reality only nearby users of the same channel will cause congestion to one another, and distantly spaced users may access the same channel without degrading each other’s QoS. This is known as spatial reuse – where the same piece of spectrum can be used by many distantly separated users without detrimental effects.

The protocol interference model [24], [25] is a commonly used model to approximate how the positions of users affect their communication performance. The idea behind the protocol interference model is to construct an interference graph, where vertices represent players (wireless users), and an undirected edge connecting two players represents that these two players are within interference range of one another (hence they can generate interference to each other if transmitting on the same channel). By using an interference graph $G$ to represent which vertices are close enough to interfere with each other, one may view the spectrum sharing problem as a game on a graph.

In this game, one may determine whether the QoS demand of a user is satisfied by counting the number of neighbors...
it has, which are using the same channel as itself. This corresponds to a generalization of the QoS satisfaction game where we account for the spatial positioning of the users.

Let us define a spatial QoS satisfaction game to be a quadruple \((\mathcal{N}, \mathcal{C}, (T_n^c)_{n \in \mathcal{N}, c \in \mathcal{C}}, G)\) where:

- \(\mathcal{N}, \mathcal{C}\), and \(T_n^c\) are the set of players/users, channels, and thresholds, respectively, which are the same as those introduced in Section II-C.
- \(G = (\mathcal{N}, \mathcal{E})\) is an undirected and unweighted graph, with a vertex set equal to the set of players \(\mathcal{N}\), and an edge set \(\mathcal{E}\). We refer to \(G\) as the interference graph. The interpretation of \(G\) is that there is an edge \((n, m) \in \mathcal{E}\) if and only if users \(n\) and \(m\) are close enough to cause congestion to each other when transmitting on the same channel. We can apply the interference estimation methods in [26], [27] to obtain the interference graph.

As before, a strategy profile \(\mathbf{x} = (x_1, x_2, \ldots, x_N)\) is where each player \(n\) chooses a strategy \(x_n \in \mathcal{C}\). Let us define the neighborhood of player \(n\), to be \(\text{Ne}(n) = \{m : (n, m) \in \mathcal{E}\} \cup \{n\}\). In other words \(\text{Ne}(n)\) is the set of all players which are linked to, or identical to \(n\). We let the neighborhood of a player contain the player itself just for the notational convenience.

Let use define the local congestion level of channel \(c\) for player \(n\) in strategy profile \(\mathbf{x}\) to be

\[ I_n^c(\mathbf{x}) = |\{m \in \text{Ne}(n) : x_m = c\}|. \]

In other words, \(I_n^c(\mathbf{x})\) denotes the number of players within a graph-distance 1 of \(n\) that are using the same channel as \(n\). The utility player \(n\) gets in strategy profile \(\mathbf{x}\) is defined in a similar way to Equation (4), from Subsection II-C. We illustrate a spatial QoS satisfaction game in Figure 6.

**Theorem 7.** Every \(N\)-players spatial QoS satisfaction game has the finite improvement property. Moreover, any asynchronous better response update process will reach a pure Nash equilibrium within \(4N + 3N^2\) asynchronous better response updates (irrespective of the initial strategy profile, or the order in which the players update).

**Theorem 7** is the most powerful result in this paper, for it implies that every spatial QoS satisfaction game, with heterogenous players and heterogenous channels has the finite improvement property. The type of QoS satisfaction games we defined in Section II can be considered as special cases of spatial QoS satisfaction games within which the interference graph is a complete graph. For this reason Theorem 1 can be considered to be a corollary of Theorem 7. Theorem 7 shows that spatial QoS satisfaction games are a remarkable class of congestion games on graphs, because they may have heterogenous channels and users, and yet they always have the finite improvement property. If one considers the slightly more general class of congestion games on graphs [8] with arbitrary non-increasing utility functions, then one can easily find example games which do not even have pure Nash equilibria -never mind the finite improvement property. For example, a congestion game on a graph with 5 players and 3 resources, without any pure Nash equilibria is exhibited in [8].

V. DISTRIBUTED ALGORITHM AND SIMULATIONS

A. Distributed QoS satisfaction algorithm

In this section we propose a distributed QoS satisfaction algorithm for achieving pure Nash equilibria of general (spatial) QoS satisfaction games. The key idea is to utilize the finite improvement property and let one user improve its channel selection at a time. In order to describe the QoS satisfaction game purely in terms of channel selection, we may regard the dominant state 0 as an addition virtual channel, which always gives users a utility of 0.

We consider a time-slotted system (see Figure 7). Each time slot \(t\) consists of the following two parts:

1) **Spectrum Access:** Each user \(n\) contends to access the chosen channel \(x_n\) according to some medium access control (MAC) mechanism. For the initialization, we assume that all users are dormant, and use strategy 0.

2) **Channel Update Contention:** We exploit the finite improvement property by having one user carry out a channel update at each time slot. In this part, we let users who can improve their channel selections compete for the channel update opportunity in a distributed manner. More specifically, each user \(n\) first computes its set of best responses (which is the set of strategies which maximize (and increase) \(n\)'s utility).

\[ \mathcal{B}_n(\mathbf{x}) = \{c^* : c^* = \text{arg max}_{c \in \mathcal{C}} U_n(c, \mathbf{x}_{-n}) \text{ and } U_n(c^*, \mathbf{x}_{-n}) > U_n(\mathbf{x})\}. \]

If \(\mathcal{B}_n(\mathbf{x}) \neq \emptyset\) (i.e., user \(n\) can improve), then user \(n\) will contend for the channel update opportunity. Otherwise, user \(n\) will not contend and will adhere to the original channel selection \(x_n\) at next time slot.

For the channel update contention, for example, we can adopt the backoff-based mechanism by setting the time length of channel update contention as \(\tau^*\). Each contending user \(n\) first generates a backoff time value \(\tau_n\) according to the uniform distribution over \([0, \tau^*]\) and waits until the backoff timer expires. When the timer expires, if the user has not
Algorithm 2: Distributed QoS satisfaction algorithm

1. **Initialization**: each user \( n \) chooses channel \( x_n = 0 \).

2. **for each user** \( n \) **and each time slot** \( t \) **do**

   3. access the chosen channel \( x_n \).

   4. compute the set of best response channel selections \( B_n(x) \).

   5. **if** \( B_n(x) \neq \emptyset \) **then**

      6. contend for the channel update opportunity.

      7. **if** win the channel update contention **then**

         8. choose a channel \( c^* \in B_n(x) \) randomly for next time slot.

         9. broadcast the updated channel selection \( c^* \) to other users.

      **else**

         10. choose the original channel \( x_n \) for next time slot.

  **else**

    12. choose the original channel \( x_n \) for next time slot.

    13. update the channel selections \( x_{-n} \) of other users once an updating message is received.

received any updating messages from other users yet, the user will randomly select a channel \( c^* \in B_n(x) \) and broadcast an updating message over the common control channel to indicate that it will update its channel selection to \( c^* \) at the beginning of the next time slot.

According to the finite improvement property in Theorem 7, the algorithm will converge to a pure Nash equilibrium of a general spatial QoS satisfaction game in polynomial time.

**B. Numerical Results**

We now evaluate the proposed distributed QoS satisfaction algorithm by simulations. We consider a spectrum sharing network of \( C = 4 \) vacant channels, with the mean data rates \( B_c \) of 6, 9, 12, 18 Mbps, respectively, which are standard operating data rates in IEEE 802.11g systems [23]. Multiple users are randomly scattered over a 100 m \( \times \) 100 m region (see Figure 8 for an illustration). In the interference graph, a pair of users are linked by an edge when they are within 50 m (the interference range) of each other (i.e., when they can generate interference to each other). We adopt the TDMA mechanism for the medium access control (MAC) and the data rate of user \( n \) choosing a channel \( c \) is given as \( I_{nc}^{(t)} = I_{nc}^{(t)}(x) \), where \( I_{nc}^{(t)}(x) \) is the number of users of channel \( c \) that are linked to \( n \) upon the interference graph. We consider the scenario where users are running two different multimedia applications corresponding to two types of QoS demands: low demand type \( D_n = 0.125 \) Mbps (i.e., listening to an online MP3 song [28]) and high demand type \( D_n = 3.5 \) Mbps (i.e., watching an online video with a resolution of 1080p [28]).

We first implement a simulation with \( N = 50 \) users, and let the fraction of users with a high QoS demand vary from 0% to 100%. We implement the distributed QoS satisfaction game solution in Algorithm 2. Figure 9 shows the dynamics of users’ throughputs, which demonstrates that the proposed distributed QoS satisfaction algorithm can converge to a pure Nash equilibrium. As a benchmark, we also compute the social optimum by the centralized optimization using Cross Entropy method, which is an advanced randomized searching technique and has been shown to be efficient in solving complex combinatorial optimization problems [29]. The results are shown in Figure 10. The x-axis is the fraction of users having a high QoS demand, and y-axis describes how many users are satisfied at the solutions of pure Nash equilibria and social optima. Note that a QoS satisfaction game may have multiple pure Nash equilibria, and Algorithm 2 will randomly select one pure Nash equilibrium (since a random user will be chosen for channel selection update). We run the algorithm 20 times for each game instance and plot the number of satisfied users at the obtained pure Nash equilibria. Figure 10 shows that both the performances of social optima and (the best and the worst) pure Nash equilibria decrease as the fraction of users of a high QoS demand increases. This is because that given the constant spectrum resources less users can be satisfied when more users have higher demands. Compared with the social optima, the performance loss by the best pure Nash equilibria and the worst pure Nash equilibria by Algorithm 2 are at most 7% and 20%, respectively (not shown in the figure). This demonstrates the efficiency of the pure Nash equilibria of QoS satisfaction games.

We implement another simulation with the number of users \( N = 50, 55, \) and 60 with half of the users having a high QoS demand. Upon comparison, we also implement the social optimum solution by centralized optimization and the decentralized spectrum access solution by Q-learning mechanism proposed in [30]. We observe that the distributed QoS satisfaction algorithm can achieve up-to 32% performance gain over the Q learning mechanism. Compared with the centralized optimization, the performance loss of the distributed QoS satisfaction algorithm is at most 10%. This demonstrates the efficiency of the proposed distributed QoS satisfaction algorithm. We next evaluate the convergence time of the distributed QoS satisfaction algorithm. Figure 12 shows that the average convergence time increases linearly with the
number of users $N$. This shows that the distributed QoS satisfaction algorithm scales well with the network size. This is critical since computing the social optimum of general QoS satisfaction games is NP-hard.

VI. CONCLUSION

In this paper, we proposed a framework of QoS satisfaction games to model the distributed QoS satisfaction problem among wireless users. The game based solution is motivated by the observation that the centralized optimization problem of maximizing the number of satisfied users is NP hard. We have explored many aspects of QoS satisfaction games including the pure Nash equilibria and the price of anarchy. Our results reveal that selfish spectrum sharing can be a very effective way to allow users to meet their QoS demands. In particular, we have shown that our systems can always reach a pure Nash equilibrium in polynomial time, simply by having the users perform better response updates.

There are many other issues we wish to explore in the future. In particular, we wish to extend many of our results (such as those regarding the price of anarchy) to spatial QoS satisfaction games. We also wish to explore the generalized QoS satisfaction games where different players receive different utilities for being satisfied.

APPENDIX

A. Proof of Theorem 7

Let us define the function $\Phi$ (which maps strategy profiles to real numbers) such that for each strategy profile $x$ we have $\Phi(x) = \left(\sum_{n \in N : x_n \neq 0} T_{x_n} - \sum_{c=1}^{C} \left( |\{n, m \in E : x_n = x_m = c\}| + |\{n \in N : x_n = c\}| \right) \right)$. Here $|\{n, m \in E : x_n = x_m = c\}|$ is the number of edges linking players using channel $c$, and $|\{n \in N : x_n = c\}|$ is the number of players using channel $c$. In other words, $\Phi(x)$ is equal to [the sum of the thresholds which the non-dormant users associate with their channels] minus [the number of edges linking users of the same channel] minus [half the number of non-dormant users].
Suppose player $n'$ does a better response update by changing their strategy from $c' \in \{0, 1, \ldots, C\}$ to $d' \in \{0, 1, \ldots, C\}$, and this has the effect of changing the strategy profile from $x$ to $y = (x_1, \ldots, x_{n'-1}, d', x_{n'+1}, \ldots, x_N)$. Next we will show $\Phi(y) \leq \Phi(x) + \frac{t}{2}$ in each of the three possible cases:

1. $c' = 0, d' \neq 0$ (i.e., when $n'$ stops being dormant).
2. $c' = 0, d' = 0$ (i.e., when $n'$ becomes dormant).
3. $c' \neq 0, d' \neq 0$ (i.e., when $n'$ switches from one channel to another).

In case 1), where $c' = 0, d' \neq 0$, we have $\Phi(y) = \Phi(x) + T_n^{d'} - T_n^{d}(x) - \frac{1}{2}$, because the action where player $n'$ switches to channel $d'$ increases the number of edges linking users of $d'$ by $I_n^{d'}(x)$ and increases the number of players using resource $d'$ by 1. Also, since our move is a better response update, we have $U_n^{d'}(x) = 0$ and $U_n^{d}(y) = 1$, and so $T_n^{d'} \geq T_n^{d}(y) = I_n^{d'}(x) + 1$. It follows that $\Phi(y) - \Phi(x) = T_n^{d'} - I_n^{d'}(x) - \frac{1}{2} \geq 1, 2$.

In case 2), where $c' \neq 0, d' = 0$ we have $\Phi(y) = \Phi(x) - T_n^{d'} + I_n^{d'}(x) - 1 + \frac{1}{2}$, because the action where player $n'$ leaves channel $c'$ decreases the number of edges linking users of $c'$ by $I_n^{c'}(x)$ and decreases the number of users of resource $c'$ by 1. Also, since our move is a better response update, we have $U_n^{c'}(x) = -1$ and $U_n^{c'}(y) = 0$. It follows that $T_n^{c'} \leq I_n^{c'}(x) - 1$. It follows that $\Phi(y) - \Phi(x) = I_n^{c'}(x) - 1 - T_n^{c'} + \frac{1}{2} \geq \frac{1}{2}.$

In case 3), where $c' \neq 0, d' \neq 0$, we have $\Phi(y) = \Phi(x) + T_{n'}^{d'} - I_{n'}^{d'}(x) - T_{n'}^{d'} + I_{n'}^{d'}(y) - 1$, because the action where player $n'$ switches from channel $c'$ to channel $d'$ decreases the number of edges linking users of $d'$ by $I_{n'}^{d'}(x)$ and decreases the number of edges linking users of $c'$ by $I_{n'}^{c'}(x) - 1$. Also, since our move is a better response update, we have $U_{n'}^{c'}(x) = -1$ and $U_{n'}^{c'}(y) = 1$, and so $T_{n'}^{d'} \geq I_{n'}^{d'}(y) = I_{n'}^{d'}(x) + 1$ and $T_{n'}^{c'} \leq I_{n'}^{c'}(x) - 1$. It follows that $\Phi(y) - \Phi(x) \geq 1.$

Without loss of generality, we can suppose that $-1 \leq T_n^{c'} \leq N + 1, \forall n \in N, \forall c \in \{1, 2, \ldots, C\}$, since thresholds less than $-1$ induce the same kind of behavior as thresholds equal to $-1$ (i.e., they can never be satisfied) and thresholds greater than $N + 1$ induce the same kind of behavior as thresholds equal to $N + 1$ (i.e., they are always satisfied). For any strategy profile $x$ we have $(1) N \leq \left( \sum_{n \in N : x_n = 0} T_n^{c_x} n \right) \leq N(N + 1).$ It is also true that $0 \leq \sum_{c \in C} \sum_{n \in N : x_n = c} I_n^{c}(x) \leq \frac{N(N-1)}{2}$ and $0 \leq \sum_{c \in C} \sum_{n \in N : x_n = c} \frac{1}{2} \leq \frac{N}{2}$. From these inequalities, it follows that $-N \leq \Phi(x) \leq N(N + 1) + \frac{C N(N-1)}{2} + \frac{C N}{2} = N + 4N^2/2.$

When we start to evolve our system, the value of $\Phi$ for the initial strategy profile cannot be less than $-N$. Also, the value of $\Phi$ will increase by at least $\frac{1}{2}$ with every better response update. Now suppose we have performed $t$ better response updates (i.e., we have run the system for $t$ time slots) and arrived at strategy profile $y$. We must have $-N + \frac{t}{2} \leq \Phi(x) + \frac{t}{2} \leq \Phi(y) \leq N + 4N^2/2$, because the value of $\Phi$ increases by at least $\frac{1}{2}$ on each time step. This implies $t \leq 4N + 3(N)^2$.

So far we have shown that it is impossible to run the system (with asynchronous better response updates) for more than $t = 4N + 3(N)^2$ time slots. This implies that when we evolve the system under asynchronous better response updates, we must reach a strategy profile $z$ from which no further better response updates can be performed, within $4N + 3(N)^2$ time slots. Such a strategy profile $z$ must be a pure Nash equilibrium by definition. □

REFERENCES


Richard Southwell received his B.Sc in theoretical physics and M.Sc in mathematics from the University of York, UK, in 2005 and 2006, respectively. He received his Ph.D degree in mathematics from Sheffield University, U.K., in 2009. Previously he worked as a post doctoral research fellow at the Department of Mathematics in the University of Sheffield in 2009. In 2010 he became a post doctoral research fellow at the Department of Information Engineering in the Chinese University of Hong Kong. In 2011 he became an Assistant Professor at the Institute for Interdisciplinary Information Sciences at Tsinghua University, Beijing.

Currently Richard is a post doctoral research fellow at the Department of Information Engineering in the Chinese University of Hong Kong. His research interests include network science, game theory and complex systems. He has served on the technical program committee for IEEE GLOBECOM 2013 and ICCVE 2013.

Xu Chen (S’10-M’12) received the B.S. degree in electronic engineering from the South China University of Technology (Guangzhou, Guangdong, China) in 2008, and the Ph.D. degree in information engineering from the Chinese University of Hong Kong (Hong Kong, China) in 2012. Dr. Chen is currently a postdoctoral research fellow in the School of Electrical, Computer and Energy Engineering, Arizona State University (Tempe, Arizona, USA). His general research interests include cognitive radio networks, wireless resource allocation, network economics, mobile social networks, and game theory. He is the recipient of the Honorable Mention Award (the first runner-up of the best paper award) in IEEE international conference on Intelligence and Security Informatics (ISI), 2010.

Jianwei Huang received his B.S. degree in Electronic Engineering from the South China University of Technology (Guangzhou, Guangdong, China) in 2008, and the Ph.D. degree in Information Engineering from the Chinese University of Hong Kong in 2012. Dr. Huang is currently a postdoctoral research fellow at the School of Electrical, Computer and Energy Engineering, Arizona State University (Tempe, Arizona, USA). His research interests include network optimization and game theoretical analysis of networks, especially on network economics, cognitive radio networks, and smart grid. He is the co-recipient of IEEE Marconi Prize Paper Award in Wireless Communications 2011, and Best Paper Awards from IEEE WiOPT 2013, IEEE SmartGridComm 2012, WiCON 2011, IEEE GLOBECOM 2010, and APCC 2009. He received the IEEE ComSoc Asia-Pacific Outstanding Young Researcher Award in 2009. Dr. Huang has served as the Editor of IEEE Journal on Selected Areas in Communications - Cognitive Radio Series, Editor of IEEE Trans. Wireless Communications, and Guest Editor of IEEE Journal on Selected Areas in Communications and IEEE Communications Magazine. He is the Chair of IEEE ComSoc Multimedia Communications Technical Committee, a Steering Committee Member of IEEE Trans. Multimedia and IEEE ICME. He has served as the TPC Co-Chair of IEEE SmartGridComm Demand Response and Dynamic Pricing Symposium 2014, IEEE GLOBECOM Selected Areas of Communications Symposium 2013, IEEE WiOpt 2012, IEEE ICCCC Communication Theory and Security Symposium 2012, IEEE GLOBECOM Wireless Communications Symposium 2010, IWCMC Mobile Computing Symposium 2010, and GameNets 2009.