# Graph-Based Robust Resource Allocation for Cognitive Radio Networks*

Lu Lu†, Dawei He‡, Xingxing Yu‡ and Geoffrey Ye Li†
† School of ECE, Georgia Institute of Technology, USA
‡ School of Math, Georgia Institute of Technology, USA

***Abstract*—In this paper, we investigate robust resource allocation for cognitive radio networks. First, a resource allocation scheme based on stable matching is developed, which takes the preferences of both secondary users and primary users into account. To improve its robustness, we then discuss an $\epsilon$-stable resource allocation scheme. With the help of the properties of $\epsilon$-stable resource allocation, three edge-cutting algorithms are proposed. Numerical results show that the modified algorithms are robust to the channel state information variation.**

***Index Terms*—Cognitive radio (CR) network, stable matching, robustness, edge-cutting**

## I. Introduction

To improve the spectrum efficiency of wireless networks, *cognitive radio* (CR) technology has been developed [1], [2], where *secondary users* (SUs) are allowed to use the licensed spectrum bands as long as they do not generate unacceptable interference to the *primary users* (PUs). Due to coexistence of PUs and SUs, appropriate resource allocation is important to better utilize spectrum.

There are different types of resource allocation schemes for CR networks. Traditional schemes aim at optimizing the sum/average utilities of SUs, such as throughput and energy efficiency, for given interference constraints to the PUs [3]–[5]. If competition feature among SUs is considered, game theory is commonly used for solving this type of problems [6]–[8]. Besides considering the SUs' performance only, the activities of PUs can also be considered jointly in the resource allocation [9] to further optimize the system. One typical scenario is spectrum trading [10], where PUs benefit from selling/renting their own spectrum bands to SUs. However, existing resource allocation schemes mainly focus on the design based on a fixed system condition while the robustness to the variation of the system condition is seldom investigated.

In this paper, we investigate robust resource allocation using graph theory. First, by taking both SUs and PUs preference lists into account, resource allocation is performed based on the Gale-Shapley stable matching algorithm. To improve the robustness of resource allocation, a truncated Gale-Shapley algorithm is studied. Upper bounds on the number of rounds needed to reach $\epsilon$-stable and $(1+\epsilon)$-approximation of stable resource allocation are derived, respectively. Motivated by our analytical results, we develop three edge-cutting algorithms to further improve the robustness of resource allocation. Numerical results show that our proposed algorithms can provide robust resource allocation results to the *channel state information* (CSI) variation while keeping the utility gap from stable resource allocation small.

*This work was supported in part by the NSF under Grant AST-1247545.

## II. System Model

We consider an underlay CR network with multiple channels/bands, where PUs have priorities to use $N$ spectrum channels/bands while $M$ SU pairs want to transmit simultaneously. All channels are modeled as Rayleigh block fading channels. Without loss of generality, we assume the $j$-th PU uses the $j$-th spectrum band. The channel between the $i$-th SU pair on the $j$-th spectrum band and the interference channel from the $i$-th SU transmitter to the $j$-th PU receiver are denoted as $h_{i,j}$ and $g_{i,j}$, respectively. The transmit power of the $i$-th SU transmitter on the $j$-th spectrum band is $P_{i,j}$. The noise power on all spectrum bands is assumed to be the same, denoted as $\sigma^2$. A centralized system is assumed, where the control center knows CSI and allocates resources based on it. For each SU, the control center may know CSI of all or a part of spectrum bands. Any SU can only be allocated on the spectrum bands that the CSI is available, including both CSI between the SU pairs and of the interference channel from the SU transmitter to the PU receiver. Correspondingly, the number of spectrum bands available for the $i$-th SU is denoted as $\Delta_{s,i}$ and the the number SUs available for the $j$-th spectrum band is denoted as $\Delta_{p,j}$. If not otherwise specified, the discussion involved CSI is only for spectrum bands and SUs that CSI is available at the control center. We consider the scenario that only one SU pair is allowed on each spectrum band and each SU pair can only access at most one spectrum band.

To protect PUs, the interference signal power generated by SUs on the $j$-th spectrum band should be below a given threshold, that is,

$$I_{i,j} = P_{i,j}|g_{i,j}|^2 \leq I_{th}, \quad \forall i, j, \tag{1}$$

where $I_{th}$ is the interference threshold[1]. Moreover, due to the amplifier capacity limit, each SU transmitter has a peak transmit power constraint, $P$, that is,

$$P_{i,j} \leq P, \quad \forall i. \tag{2}$$

When performing resource allocation to SUs, we jointly consider benefits of PUs by incorporating the concept of

[1]Without loss of generality, we assume the interference threshold is same on all the spectrum bands for simplicity. The results can be directly extended to the system with different interference thresholds on different spectrum bands.

spectrum trading into the utility function design [10]. PUs will charge more from SUs if better performance is provided to SUs. On the other hand, performance of PUs will degrade if SUs generate strong interference. Therefore, while SUs improve their own performance by increasing their transmit powers, they should also get penalties for generating stronger interference. To capture the features of both PUs and SUs, we will use the following utility function,

$$w_{i,j}(P_{i,j}) = c_s \log_2\left(1 + \frac{|h_{i,j}|^2 P_{i,j}}{\sigma^2 + I^p_{j,i}}\right) - c_p P_{i,j}|g_{i,j}|^2, \quad (3)$$

if the $i$-th SU uses the $j$-th spectrum band, where $c_s$ and $c_p$ are weight factors. $I^p_{j,i}$ denotes the interference from the $j$-th PU to the $i$-th SU receiver. We assume the interference powers, $I^p_{j,i}$, are known since the PUs' transmit powers keep same with and without SUs' transmission and then, $I^p_{j,i}$ can be estimated in advance. Even though we will use utility in (3) for resource allocation in this paper, the developed approaches can be also used for other utility functions.

The transmit power, $P_{i,j}$, can be optimized to maximize the utility function subject to constraints in (1) and (2). Since the utility function in (3) is a concave function of $P_{i,j}$, the optimal transmit power, $P^*_{i,j}$, is

$$P^*_{i,j} = \left(\min\left\{\frac{c_s}{c_p|g_{i,j}|^2} - \frac{\sigma^2 + I^p_{j,i}}{|h_{i,j}|^2}, P, \frac{I_{th}}{|g_{i,j}|^2}\right\}\right)^+, \quad (4)$$

where $(x)^+ = \max\{0, x\}$. Then, if the $i$-th SU is allocated on the $j$-th spectrum band to transmit, the utility value of the $i$-th SU and the $j$-th PU can be expressed as $w_{i,j}(P^*_{i,j})$.

## III. Resource Allocation Based on Stable Matching

As indicated before, we will focus on resource allocation based on graph theory by taking both PUs' and SUs' preferences into account. We first describe our resource allocation scheme based on stable matching, which has similar idea as in [11], and then discuss properties of the proposed stable resource allocation scheme.

As shown in Fig. 1, we start with constructing a bipartite graph with bipartitions $V_1, V_2$, where nodes in $V_1$ represent SU pairs and nodes in $V_2$ represent spectrum bands/PUs. A edge is put between the $i$-th SU in $V_1$ and the $j$-th PU in $V_2$ if the CSI of the $i$-th SU on the $j$-th spectrum band is known at the control center. Then, the common utility value of the $i$-th SU and the $j$-th PU, $w_{i,j}(P^*_{i,j})$, is associated with the corresponding edge.

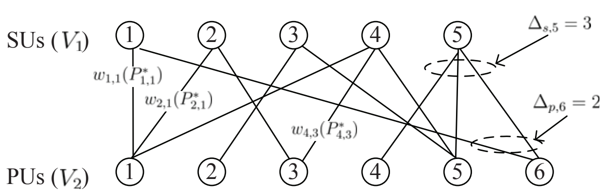


Fig. 1. Bipartite graph illustration with $M = 5$ and $N = 6$.

We then define preference lists for all SUs and PUs. For completeness, set $w_{i,j}(P^*_{i,j}) = 0$ if the CSI of the $i$-th SU on the $j$-th spectrum band is not available. For the $i$-th SU, its preference list is defined as

$$\mathbf{L}^s_i = [j^s_1, ..., j^s_N], \quad (5)$$

where $j^s_1, ..., j^s_N$ is a permutation of $1, ..., N$ satisfying

$$w_{i,j^s_1}(P^*_{i,j^s_1}) \leq ... \leq w_{i,j^s_N}(P^*_{i,j^s_N}). \quad (6)$$

Similarly, the $j$-th PU's preference list is defined as

$$\mathbf{L}^p_j = [i^p_1, ..., i^p_M], \quad (7)$$

where $i^p_1, ..., i^s_M$ is a permutation of $1, ..., M$ satisfying

$$w_{i^p_1,j}(P^*_{i^p_1,j}) \leq ... \leq w_{i^p_M,j}(P^*_{i^p_M,j}). \quad (8)$$

We call the $i$-th SU and the $j$-th PU is a matched pair if the $i$-th SU pair is assigned to transmit on the $j$-th spectrum band, denoted as $(i, j)$. Two matched pairs $(i, k)$ and $(t, j)$ ($i \neq t$ and $k \neq j$) are unstable if the $i$-th SU prefers the $j$-th spectrum band to the $k$-th spectrum band while the $j$-th PU prefers the $i$-th SU to the $t$-th SU, i.e., $w_{i,j} > w_{i,k}$ and $w_{i,j} > w_{t,j}$; the pairs $(i, k)$ and $(t, j)$ are unstable because the pair $(i, j)$ improves the preferences of both $i$-th SU and $j$-th PU.

Based on the above definitions, we can find a stable resource allocation, i.e., no unstable matched pair, by using the Gale-Shapley algorithm [12]. The Gale-Shapley algorithm runs many rounds to get stable resource allocation, where each round consists of a proposing and a answering procedure. It can be operated in two different ways: the PU-proposing way where PUs propose to SUs and the SU-proposing way where SUs propose to PUs. The Gale-Shapley algorithm is originally developed for the case that the number of SUs equals the number of spectrum bands ,i.e., $M = N$. It can be easily generated to the case when $M \neq N$. For example, when the number of SUs exceeds the number of spectrum bands, i.e., $M > N$, we can add $(M - N)$ virtual spectrum bands/PUs and put them at the end of SUs' preference lists. Virtual PUs' preference lists can be constructed randomly. Based on the stable matching results, the SUs assigned on the original $N$ spectrum bands can transmit while the SUs assigned on the virtual spectrum bands cannot. We can treat the issue similarly when the number of SUs is less than the number of PUs.

Given one stable resource allocation, denoted as $\mathcal{S}$, its utility, denoted as $w_{\mathcal{S}}$, is defined as the sum utilities of SUs/PUs based on it. Note that the sum utilities of SUs and PUs based on a given resource allocation are the same since a common utility is used for any matched pairs. For the proposed resource allocation, we have the following theorems.

**Theorem 1.** *Based on our operations, the final stable resource allocation has maximum sum/average weight among all possible stable resource allocations. Moreover, the utilities of the PU-proposing and the SU-proposing schemes are the same.*

Due to the space limitation, we omit the proofs. Note that, even though the final stable resource allocation of the PU-proposing and the SU-proposing schemes have the same sum/average utility, the resource allocation results may different. In the PU-proposing algorithm, each PU is matched to the best possibility and each SU is matched to the worst possibility among all stable matchings; in the SU-proposing algorithm, vice versa. We can choose one of them according to the practical requirements. To simplify the discussion, we focus on the PU-proposing algorithm in the rest of the paper. The stable resource allocation and its utility based on the PU-proposing Gale-Shapley algorithm are denoted as $\mathcal{G}$ and $w_{\mathcal{G}}$, respectively.

## IV. Almost Stable Resource Allocation

In the previous section, we propose a stable resource allocation scheme for given preference lists based on the Gale-Shapley algorithm and discuss its properties. However, the stable matching results are not robust to CSI variation. CSI variation of one channel may lead to a totally different stable matching result, which is not preferable practically.

To increase the robustness of the resource allocation, a truncated Gale-Shapley algorithm is considered here, which increases the robustness of resource allocation by allowing some unstable matched pairs. Instead of running the Gale-Shapley algorithm for many rounds to get stable resource allocation, the truncated Gale-Shapley algorithm outputs a resource allocation result after a fixed number of rounds. Denote the number of rounds when the algorithm is terminated as $T$. From [13], based on the truncated Gale-Shapley algorithm, a change of CSI of one node will only affect the part of resource allocation $\mathcal{M}$ within distance $2T$ from the node of the change. Comparing with the stable resource allocation where a change of CSI of one node may affect all nodes in the system, allocation based on the truncated Gale-Shapley algorithm is much more robust to the CSI variation.

Based on the truncated Gale-Shapley algorithm, the resulting resource allocation may not be stable. To measure the stability of a resource allocation result, the concept called as $\epsilon$-stable, or almost stable, is introduced. A resource allocation $\mathcal{M}$ is called as an $\epsilon$-stable resource allocation if the number of unstable matched pairs in $\mathcal{M}$ is at most $\epsilon|\mathcal{M}|$, where $|\mathcal{M}|$ is the number of matched pairs. From [13], to find an $\epsilon$-stable resource allocation, the number of required rounds, $T$, is upper bounded by $2 + \Delta^2/\epsilon$, where $\Delta$ is the maximum number of edges connected to any of SUs and PUs, i.e., $\Delta = \max\{\Delta_{s,i}, \Delta_{p,j}\}$, $\forall i, j$. Based on the procedure in [13], we can further restrict the upper bound in the following theorem by using the maximum number of edges connected to any of SUs, i.e., $\Delta_s = \max\{\Delta_{s,i}\}$, instead of both SUs and PUs.

**Theorem 2.** *We can find an $\epsilon$-stable resource allocation in $T$ rounds, where $T \leq 2 + \Delta_s^2/\epsilon$ and $\Delta_s = \max\{\Delta_{s,i}\}$.*

Besides the stability property, we also concern about the utility of resource allocation. A resource allocation, $\mathcal{M}$, is called as a $(1+\epsilon)$-approximation of the maximum-weight stable matching if its utility $w_{\mathcal{M}}$ satisfying $(1+\epsilon)w_{\mathcal{M}} \geq w_{\mathcal{G}}$, where $\mathcal{G}$ is the stable resource allocation based the Gale-Shapley algorithm and $w_{\mathcal{G}}$ is the corresponding utility. For the truncated Gale-Shapley algorithm, we have the following conclusion regarding to its utility.

**Theorem 3.** *We can find a $(1+\epsilon)$-approximation of the maximum-weight stable resource allocation in $T$ rounds, where $T \leq 2 + \Delta_s/\epsilon$.*

There is another conclusion regarding to the utility based the truncated Gale-Shapley algorithm in [13], which compares the result with the maximum-weight resource allocation instead of stable resource allocation. In our studied scenario, it is more reasonable to compare with other stable resource allocation.

## V. Edge-Cutting for Robustness Design

From the discussions in the last section, the robustness of resource allocation based on the truncated Gale-Shapley algorithm depends on the number of rounds it runs before termination, $T$. Smaller $T$ leads to higher robustness, and vice versa. Since the exact number of $T$ satisfying $\epsilon$-stable and $(1+\epsilon)$-approximation depends on the instantaneous CSI, we focus on reducing the upper bounds of $T$ instead. From Theorems 2 and 3, the upper bounds of $T$ for satisfying $\epsilon$-stable and $(1+\epsilon)$-approximation are both related to the maximum number of available bands of any SU, $\Delta_s$. Both upper bounds can be decreased by decreasing $\Delta_s$. Thus, for improving robustness of resource allocation, a small $\Delta_s$ is preferable. It is possible that SUs only access CSI of a small number of PU bands, and $\Delta_s$ is small naturally. However, PUs, in order to improve their own utilities, may be willing to share their information with SUs to attract more SUs, as in the spectrum trading scenario. In this case, the maximum number of bands available at SUs, $\Delta_s$, may be large. Note that, the number of available bands for the $i$-th SU is the number of edges connected to it in the constructed bipartition graph. For robustness design, we propose three edge-cutting algorithms to eliminate the maximum number of edges connected to SUs, $\Delta_s$. In other words, if the number of available bands at the $i$-th SU, $\Delta_{s,i}$, is larger than a given threshold, denoted as $\Delta_c^{max}$, it will be asked to give up some bands before running resource allocation schemes.

One way to reduce the maximum number of edges connected to SUs, $\Delta_s$, is to eliminate edges based on SUs' preference lists. If the required maximum number of edges connected to SUs after edge-cutting is $\Delta_c^{max}$, each SU can keep $\Delta_c^{max}$ edges on the top of their preference lists. This is called as SU-preferred edge-cutting algorithm.

On the other hand, the edge-cutting can be done based on PUs' preference lists. We first provide a preference value on the edge between the $i$-th SU and the $j$-th PU, denoted as $t_{i,j}$. If the $i$-th SU is the $k$-th element on the $j$-th PU's preference list, set $t_{i,j} = k$. Then, for any SU, it can keep $\Delta_c^{max}$ edges that have highest preference values. For the edges that have same preference value, we can choose them randomly. Thus, those edges on the top of PUs' preference lists will be kept. This approach is called as PU-preferred edge-cutting algorithm. Note that, this approach also eliminates the maximum degree of all SUs, not PUs.

Besides the edge-cutting based on either SUs' or PUs' preference lists, it can be also done based on the preference lists of both sides, which is called as a double-preferred edge-cutting algorithm. We set a preference value for each edge. For the edge between the $i$-th SU and the $j$-th PU, set $t_{i,j}^p = k$

if the $i$-th SU is the $k$-th element on the $j$-th PU's preference list and set $t^s_{i,j} = l$ if the $j$-th PU is the $l$-th element on the $i$-th SU's preference list. Then, we set a preference value on the edge between the $i$-th SU and the $j$-th PU as the weighted sum of $t^p_{i,j}$ and $t^s_{i,j}$, which can be expressed as $t_{i,j} = p_s t^s_{i,j} + p_p t^p_{i,j}$, where $p_s$ and $p_p$ are positive weight factors that satisfy $p_s + p_p = 1$. Based on the preference value, at any SU, $\Delta^{max}_c$ edges that have highest preference values will be kept. The weight factors can be adjusted according to the priorities of SUs' and PUs' preference lists. When $p_s = 1$ and $p_p = 0$, this is equivalent to a SU-preferred edge-cutting while it is a PU-preferred edge-cutting when $p_s = 0$ and $p_p = 1$.

Note that, in addition to improving the robustness of resource allocation, edge-cutting also reduces computational complexity.

## VI. Numerical Results

In this section, numerical results are presented to demonstrate the performance of the proposed algorithms. Here, we consider the case with 200 SUs and 200 PUs, i.e., $M = N = 200$ and assume all CSI is known. Except stable resource allocation, all other algorithms are truncated when the corresponding resource allocation satisfying $\epsilon$-stable and its an $(1+\epsilon)$-approximation of their corresponding bipartite graphs. Results are averaged by 20,000 trails. For each trail, algorithms are conducted once based on the original CSI and then, conducted another time by changing CSI of 5 SUs. For a resource allocation scheme, smaller resource allocation variation means higher robustness. All figures show relative results compared to the stable resource allocation without edge-cutting. For all results, we set the *signal-to-noise ratios* (SNRs) between any SU pair and any interference channel from SU transmitter to the PU receiver as $-10dB$ and $-15dB$, respectively. Interference threshold is $-3dB$ and the maximum transmit power is $10dB$.

Fig. 2 shows the impact of $\epsilon$ on the utility and the SU allocation variation performance. The maximum available bands for each SU after edge-cutting is set to be $\Delta^{max}_c = 20$. Comparing with the stable resource allocation, the edge-cutting algorithms can decrease SU allocation variations by $45\%$ while keeping $0.05$-stable and $1.05$-approximation of the stable resource allocation. Comparing three edge-cutting algorithms, the SU-preferred algorithm has lowest SU allocation variation, that is, it has highest robustness, while it has largest utility gap compared to stable resource allocation. On the other hand, the PU-preferred algorithm is the least robust one while it has smallest utility gap from stable resource allocation. The double-preferred edge-cutting algorithm can provide a tradeoff result.

Fig. 3 shows the impact of the maximum number of available bands for each SU after edge-cutting, $\Delta^{max}_c$, on the performance of the utility and the SU allocation variation, where $\epsilon = 0.01$. The decrease of $\Delta^{max}_c$ increases the utility gap from stable resource allocation while it increases the robustness as well. Practically, a suitable maximum number of available bands for each SU, $\Delta^{max}_c$, can be chosen to

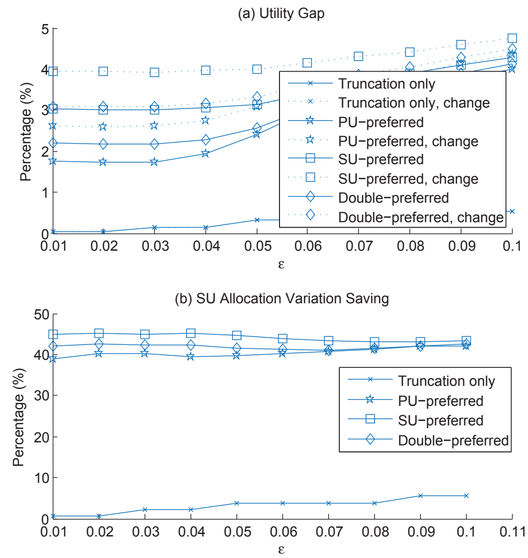


Fig. 2. The impact of $\epsilon$.

satisfy the utility requirement while keeping the robustness of resource allocation.

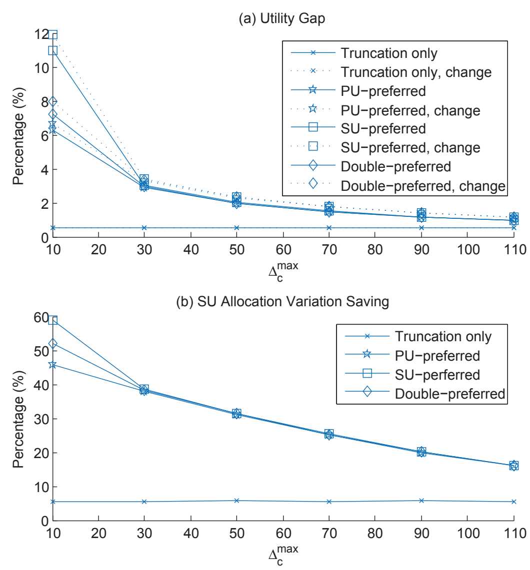


Fig. 3. The impact of the maximum number of available bands for each SU after edge-cutting, $\Delta^{max}_c$.

## VII. Conclusions

In this paper, we study robust resource allocation for CR networks. First, we develop a resource allocation scheme based on stable matching, which takes both SUs' and PUs' preferences into account. We then discuss the properties of almost stable resource allocation, which is more robust to the CSI variation than stable resource allocation. Based on the properties on almost stable resource allocation, we propose three edge-cutting algorithms to further improve the robustness of the resource allocation scheme. Numerical results show our proposed schemes are robust to the CSI variations.

## References


[1] J. Mitola and G. Q. Maguire, "Cognitive radio: making software radios more personal," *IEEE Personal Commun.*, vol. 6, no. 4, p. 13, Aug. 1999.

[2] L. Lu, X. Zhou, U. Onunkwo, and G. Y. Li, "Ten years of research in spectrum sensing and sharing in cognitive radio," *Eurasip J. Wireless Commun. and Networking*, vol. 28, Jan. 2012.

[3] L. Musavian and S. Aissa, "Capacity and power allocation for spectrum-sharing communications in fading channels," *IEEE Trans. Wireless Commun.*, vol. 8, pp. 148–156, Jan. 2009.

[4] V. Asghari and S. Aissa, "Adaptive rate and power transmission in spectrum-sharing systems," *IEEE Trans. Wireless Commun.*, vol. 9, pp. 3272–3280, Oct. 2010.

[5] X. Zhou, G. Y. Li, D. Li, D. Wang, and A. C. K. Soong, "Probabilistic resource allocation for opportunistic spectrum access," *IEEE Trans. Wireless Commun.*, vol. 9, pp. 2870–2879, Sept. 2010.

[6] Y.-E. Lin, K.-H. Liu, and H.-Y. Hsieh, "Design of power control protocols for spectrum sharing in cognitive radio networks: A game-theoretic perspective," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2010.

[7] S. Buzzi and D. Saturnino, "A game-theoretic approach to energy-efficient power control and receiver design in cognitive CDMA wireless networks," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 1, pp. 137–150, Feb. 2011.

[8] Z. Tian, G. Leus, and V. Lottici, "Joint dynamic resource allocation and waveform adaptation for cognitive networks," *IEEE J. Sel. Areas Commun.*, vol. 29, pp. 443–454, Feb. 2011.

[9] S. K. Jayaweera, G. Vazquez-Vilar, and C. Mosquera, "Dynamic spectrum leasing: A new paradigm for spectrum sharing in cognitive radio networks," *IEEE Trans. Veh. Technol.*, vol. 59, no. 5, pp. 2328–2339, June 2010.

[10] C. Xiong, G. Y. Li, L. Lu, D. Feng, Z. Ding, and H. Mitchell, "Spectrum trading for efficient spectrum utilization," *EAI Trans. Wireless Spectrum*, to appear 2014.

[11] L. Lu, D. He, X. Yu, and G. Y. Li, "Energy efficient resource allocation for cognitive radio networks," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM'13)*, 2013.

[12] D. B. West, *Introduction to Graph Theory*, 2nd ed. Prentice Hall, 2001.

[13] P. Floreen, P. Kaski, V. Polishchuk, and J. Suomela, "Almost stable matchings by truncating the gale-shapley algorithm," *Algorithmica*, vol. 28, no. 1, pp. 102–118, 2010.