This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE Globecom 2010 proceedings.

Resource Allocation with Load Balancing for Cognitive Radio Networks

Huahui Wang Jian Ren Tongtong Li
Department of Electrical & Computer Engineering
Michigan State University, East Lansing, Michigan 48824, USA
Email: wanghuah,renjian,tongli@msu.edu

Abstract— **This paper considers channel and power allocation for cognitive radio (CR) networks. We assume that the total available spectrum is divided into several bands, each consisting of a group of channels. A centralized base station, enabled by spectrum sensing, is assumed to have the knowledge of all vacant channels, which will be assigned to various CRs according to their requests. The objective of resource allocation is to maximize the sum data rate of all CRs. Since the activities of primary users may cause heavy traffic in some bands while leaving other bands idle, load balancing is first performed to equalize the traffic. A multi-level subset sum algorithm as well as a simpler greedy algorithm is proposed to achieve excellent load balancing performance. After that, an algorithm incorporated with constant-power water filling is proposed to maximize the sum data rate. Simulation results are presented to illustrate the effectiveness of the proposed algorithms.**

Index Terms: cognitive radio, load balancing, channel allocation, power allocation

I. Introduction

With the rapid development of wireless communications and the increasing demands for spectrum resources, spectrum scarcity has become a more and more serious problem. Recent studies by the FCC, however, show that the shortage of spectrum is mainly due to inefficient usage of spectrum rather than spectrum scarcity [1]. In order to improve spectrum utilization, policy reform is under way to allow unlicensed (secondary) users to opportunistically access licensed spectrum, at the premise of not causing interfere to licensed (primary) users. Such spectrum sharing is enabled by the cognitive radio (CR) technology [2], [3], which can detect and access the vacant spectrum, such that the efficiency of the spectrum utilization is improved.

As a newly emerging technology, there are many design aspects remained open for CR networks. One of the major challenges, which is the focus of this paper, is to efficiently allocate channels and power for spectrum sharing. We consider a centralized setup consisting of a base station controlling multiple CRs. Similar system is considered in [4], where joint channel and power allocation is found iteratively to maximize the sum capacity. A multi-stage algorithm is proposed in [5] for subcarrier and power allocation in the orthogonal frequency division multiple access (OFDMA)-based CR system. In [6], channel and power allocation is considered for a cellular CR network. A heuristic two-phase resource allocation scheme is proposed, where channels and power are first allocated to the base stations to maximize their total coverage area, and then each base station allocates the channels to maximize the total number of served CRs.

In most literature on resource allocation, including those mentioned above, a common assumption is that the instantaneous channel gains for those vacant channels are available for each CR. Although this is possible for "local" allocation, it may not be practical when the channels spread across a large spectrum. In [7], [8], it has been pointed out that tracking channel gains instantaneously for dynamic spectrum allocation could be very difficult in practice, and only mean channel gains averaged over short-term fading are assumed available.

In this paper, resource allocation for CR networks is considered over a large spectrum which is logically divided into several bands. Each band consists of a group of channels. We assume that instantaneous channel gains are available only when allocation is performed within "local" bands, but mean channel gains can always be available for all CRs. Load balancing is first performed to distribute CRs' traffic over the spectrum. By doing this, we introduce certain degree of flexibility and fairness in spectrum sharing: the scenarios that some parts of the spectrum are severely congested yet some are under-utilized are prevented. More importantly, such a balanced flexibility provides a capability to deal with local capacity requirements in emergency situations [9]. After load balancing, each CR is associated with a certain band. An algorithm incorporated with constant-power water filling is proposed to allocate channels and power, such that the sum data rate is maximized.

The paper is organized as follows. Section II describes the system model. Section III presents the load balancing algorithms, and Section IV is on channel and power allocation after load balancing. Simulation examples are provided in Section V and we conclude in Section VI.

II. System Model

Consider a wireless CR network with a centralized base station and N_u secondary users. Each user has a data rate request and a power constraint. The spectrum is divided into b bands, each consisting of a group of equal-sized channels of bandwidth B. The channel impulse responses are assumed to be independent to each other. The number of channels in these bands is denoted as $\boldsymbol{N_c} = [N_1, N_2, \cdots, N_b]$.

The spectrum can be allocated to CRs only when the primary users are absent. It is assumed that the base station has the global knowledge about which channels are occupied by the primary users and which ones are vacant. For $i = 1, \cdots, b$, We define the effective load factor η_i as below to measure the utilization of band i,

$$\eta_i = \frac{\text{active traffic over band } i}{\text{total capacity of band } i}. \quad (1)$$

In the paper, we consider the frequency division multiple access (FDMA) scheme, where η_i is equivalent to the ratio between the number of channels occupied by the primary users and the total number of channels in band i.

Through a dedicated feedback channel, CR users provide to the base station information about their maximum transmit power as well as desired data rates, according to which the base station will make the resource allocation. Since the number of available channels varies dynamically in CR networks, resource allocation may not guarantee the full satisfaction of users' rate requests.

For a large spectrum consisting of many channels, it is unrealistic to assume that all the instantaneous channel gains are known for each CR. However, when the channels are i.i.d, it is not difficult to obtain the mean channel power. Let H_{ki} denote the ith channel's gain for user k, and assume $E[|H_{ki}|^2]$ is known. Define

$$\gamma_{ki} = \frac{|H_{ki}|^2}{N_0 B}, \quad (2)$$

where N_0 is the one-sided noise power spectral density.

Suppose user k has a maximum transmit power P_k. If N channels are allocated to user k, then the ergodic capacity for user k is given by

$$C_k = \sum_{i=1}^{N} E[B \log_2(1 + P_{ki}\gamma_{ki})], \quad (3)$$

where P_{ki} is user k's distributed power on channel i, and $\sum_{i=1}^{N} P_{ki} = P_k$.

According to the Jensen's inequality, we can derive the upper bound of the capacity for user k as below.

$$\begin{aligned} C_k &\leq \sum_{i=1}^{N} B \log_2(1 + P_{ki}\bar{\gamma}_{ki}) \quad (4) \\ &\leq NB \log_2(1 + \frac{P_k \bar{\gamma}_{ki}}{N}), \end{aligned}$$

where $\bar{\gamma}_{ki} = E[\gamma_{ki}]$.

Consequently, when user k has a desired rate R_k, we can convert it as a request for the minimum number of channels $N_{min,k}$ where

$$N_{min,k} = \arg\min_{N} \{NB \log_2(1 + \frac{P_k \bar{\gamma}_{ki}}{N}) \geq R_k\}. \quad (5)$$

If the system has an engineering limit on the maximum number N_{limit} of channels that one user can use, then the number of channels that the system will allocate to user k is

$$M_k = \min\{N_{min,k}, N_{limit}\}. \quad (6)$$

The rate requests of N_u users will then be converted into channel requests denoted as $\boldsymbol{M_r} = [M_1, M_2, \cdots, M_{N_u}]$.

III. LOAD BALANCING

The base station has the knowledge of the effective load factors for all bands, denoted as $\boldsymbol{\eta} = [\eta_1, \eta_2, \cdots, \eta_b]$. Before allocating specific channels to users, we first associate each user to a certain band based on a load balancing criterion. The band with a large load factor will be considered as having heavy bandwidth utilization, and will be allocated to light-traffic users and/or a smaller number of users, such that the bandwidth utilization level of each band is as close to each other as possible.

A. Optimal load balancing

If the users can transmit or receive signals simultaneously across multiple bands, optimal load balancing can be achieved using an algorithm which resembles water filling, as illustrated in Fig. 1.

The water level can be determined by the algorithm in Table I. The bands are sorted in ascending order by load factor η. Denote the vector of sorted load factor as $\boldsymbol{\eta}_{sort}$. The band with lowest η, i.e., $\boldsymbol{\eta}_{sort}(1)$, is filled with user loads first. The water filling stops when all the requests are assigned to the band yet $\boldsymbol{\eta}_{sort}(1) < \boldsymbol{\eta}_{sort}(2)$. Otherwise, it pauses when the updated $\boldsymbol{\eta}_{sort}(1) = \boldsymbol{\eta}_{sort}(2)$, and then treats the two bands of the same load factor as an equivalent large band and continues the process. The shaded area in Fig. 1 represents the total traffic loads.

The optimal load balancing, however, has a strong assumption that users are able to communicate simultaneously over multiple channels separated distantly in frequencies, which is difficult to realize in practical systems. In what follows, we propose load balancing algorithms under a more reasonable assumption: users can adjust their transmission parameters from band to band in different time periods, yet at a certain moment they can only transmit over multiple channels within one band. For simplicity, we neglect the situations that one user can occupy edge channels across two adjacent bands. Two algorithms of different complexity are proposed in the following subsections under this assumption.

B. Multi-level subset sum algorithm

In this algorithm, the problem is formulated into a special multiple-knapsack problem named multiple subset sum problem (MSSP) [10], which can be described as: given a set of n users and m bands, each user i having a positive integer weight (channel request) w_i, and each band j having a positive integer capacity c_j (available channels for allocation), select a subset of users of maximum total request that can be assigned to the bands. Defining x_{ij} as the indicator that user i is associated with band j, the problem is mathematically characterized as $\forall i \in \{1, 2, \cdots, n\}$, and $\forall j \in \{1, 2, \cdots, m\}$,

$$\max \quad \sum_{j=1}^{m} \sum_{i=1}^{n} w_i x_{ij}; \quad (7)$$

$$s.t. \quad \sum_{i=1}^{n} w_i x_{ij} \leq c_j; \quad (8)$$

$$s.t. \quad x_{ij} = 0 \text{ or } 1. \quad (9)$$

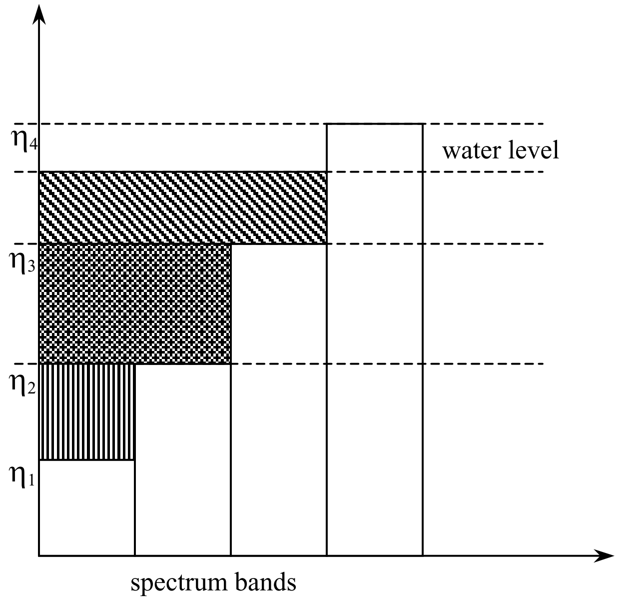

Fig. 1. Load balance using water filling.

TABLE I

SEARCH OF WATER LEVEL

1. Sort $\boldsymbol{\eta}$ in ascending order. Name it $\boldsymbol{\eta}_{sort}$.
   Obtain the corresponding sorted $\boldsymbol{N_c}$ named $\boldsymbol{N_{c,sort}}$.
2. Calculate the total user requests: $L = Sum\ \boldsymbol{M}_r(1:N_u)$.
   Initialize $waterLevel = 0$.
3. for $i = 1$ to $b-1$

   Calculate the capacity under level $\boldsymbol{\eta}_{sort}(i+1)$:
   $C = \sum_{j=1}^{j=i} N_{c,sort}(j)[\boldsymbol{\eta}_{sort}(i+1) - \boldsymbol{\eta}_{sort}(j)]$;

   if $L \le C$, i.e., all requests can be accommodated under water level $\boldsymbol{\eta}_{sort}(i+1)$, then

   $waterLevel = \frac{\boldsymbol{N}_{c,sort}(1:i)\boldsymbol{\eta}_{sort}^T(1:i)+L}{Sum\ \boldsymbol{N}_{c,sort}(1:i)}$;

   break;

   end

   end
4. if $waterLevel == 0$, i.e., L in Step 2 is too large such that Step 3 does not return a positive water level, then

   $waterLevel = \min\{1, \frac{\mathbf{N}_c\boldsymbol{\eta}^T}{Sum\ \boldsymbol{N}_c(1:b)}\}$.

   end

Dynamic programming [11] can be used to obtain the exact solution in the case of $m = 1$, i.e., single subset sum problem. However, general MSSP problems are NP-hard and approximation algorithms are usually employed for near optimal solutions. An approximation method we adopt here is called successive knapsack algorithm [12], which, as the name infers, is to run through all the bands and solve m single subset sum problems using dynamic programming.

A critical parameter for MSSP in the load balancing framework is the capacity c_j of each band. The maximum capacity for band j is determined by its total number of channels $N_c(j)$ and the load factor η_j. Specifically, $c_j = N_c(j)(1-\eta_j)$. However, if we solve the MSSP problem with maximum capacity of each band, load balancing cannot be achieved when the total amount of load requests is small. In such scenarios, successive subset sum algorithm will result in one or more bands being filled up and others left untouched. Therefore, appropriate limits on c_j should be imposed on the MSSP problem (7-9) to achieve load balancing.

A multilevel subset sum algorithm is then proposed in Table II, where we look for an appropriate level of load factor η through a process of binary search, such that the load balancing is impaired when the level is increased, and the accommodated traffic is not maximized when the level is reduced. The upper bound of the searching level is simply 1, while for the lower bound we use the searched water level obtained from the algorithm presented in Table I. The algorithm converges when the difference of two consecutive searching levels is less than a resolution defined as $1/max(\boldsymbol{N}_c)$, which is adopted by considering the constraint that the change of channel number should be no less than 1 when adjusting the levels.

TABLE II

MULTI-LEVEL SUBSET SUM ALGORITHM THROUGH BINARY SEARCH

1. Find the initial $waterLevel$ according to algorithm I,
   and let $low = waterLevel$. Set $high = 1$, and $center = low$.
2. if $low == 1$, i.e., the volume of total requests is high
   Perform successive subset sum allocation under water level 1.
   Done!
   end
3. otherwise,
   while $high - low > resolution$

   Perform successive subset sum allocation under water level $center$.

   if all users' requests can be accommodated,

   decrease water level by letting $high = center$;

   else

   increase water level by letting $low = center$;

   end

   update $center = (low + high)/2$.

   end

C. Greedy algorithm

The multi-level subset sum algorithm can achieve good load balancing performance. However, its complexity is high due to the necessity of multiple-time dynamic programming. In order to reduce complexity, a heuristic greedy algorithm is proposed, as presented in Table III.

The idea is to sort users' requests in descending order, and then sort the load factor of each band in reverse order. Requests of large amount will be assigned to the bands with small load factors. After each assignment, the load factors are updated and sorted again. The major complexity of the algorithm is in the initial sorting of users' requests and the continuous sorting of bands after each update. Since the number of bands is generally small, the complexity of sorting bands is low. Despite its simplicity, the performance of the greedy algorithm is surprisingly good, as will be shown in Section V.

IV. CHANNEL AND POWER ALLOCATION

After load balancing, each user is associated with a certain band. The problem reduces to assigning channels within one

TABLE III
GREEDY ALGORITHM

1. Sort users' requests $\boldsymbol{M_r}$ in descending order to obtain $\boldsymbol{M_{r,sort}}$.
2. Sort $\boldsymbol{\eta}$ in ascending order to get $\boldsymbol{\eta_{sort}}$, and then obtain the corresponding $\boldsymbol{N_{c,sort}}$.
3. Calculate residual capacity for each band: $\boldsymbol{C}(j) = \boldsymbol{N_{c,sort}}(j)[1 - \boldsymbol{\eta_{sort}}(j)]$, $\forall j = 1, \cdots, b$.
4. for $i = 1$ to N_u

```
     for j = 1 to b
          if M_{r,sort}(i) < C(j)
               allocate the user of request M_{r,sort}(i) to
               the band of residual capacity C(j).
               update C and η.
               re-sort η and obtain an updated η_sort.
          end
          break;
     end
end
```

band to different users such that the overall throughput is maximized. At this stage, we assume that "local" channel estimation can be performed by users and instantaneous channel gains are available.

Suppose there are L vacant channels within one band and K users associated with the band. User k will be assigned L_k channels and $\sum_{k=1}^{K} L_k \leq L$. Let H_{ki} idenote the i's channel gain of user k, and $\gamma_{ki} = |H_{ki}|^2/(N_0 B)$. User k has a power limit of P_k. The objective is to maximize the overall data rate under each user's power constraint, i.e.,

$$\max \quad R = \sum_{k=1}^{K} \sum_{i=1}^{L} r_{ki}; \tag{10}$$

$$\text{where} \quad r_{ki} = w_{ki} B \log_2(1 + P_{ki}\gamma_{ki}); \tag{11}$$

$$s.t. \;\; P_{ki} \geq 0; \tag{12}$$

$$s.t. \;\; w_{ki} = 0; \tag{13}$$

$$s.t. \;\; \sum_{i=1}^{L} P_{ki} w_{ki} \leq P_k; \tag{14}$$

$$s.t. \;\; \sum_{i=1}^{L} w_{ki} = L_k, \tag{15}$$

where w_{ki} is a binary indicator showing whether channel i is assigned to user k, and P_{ki} is user k's distributed power on channel i.

Without (15), the problem reduces to a classical sum rate maximization problem under FDMA constraint, where the optimal solution can be obtained using water filling technique [13]. However, suboptimal algorithms are more favorable in practice since they offer comparable solutions at much less cost. Constant power water filling [14], [15], among them, has been shown to have very close performance to the optimum. In the algorithm, when an optimal number of channels for one user is determined, the user's total power is equally distributed into these channels. Iterations are generally needed to obtain the optimal number of channels.

The extra constraint (15) makes the constant power water filling algorithm a superb candidate for solving our problem. Since it explicitly gives the number of channels for allocation, there is no need to iteratively calculate the number as the original algorithm does. Under constraint (15), the power allocated to each channel of user k is simply $\frac{P_k}{L_k}$, and the remaining problem is to assign channels under such power distributions to maximize the sum rate.

A proposed algorithm for channel allocation under constant power water filling is presented in Table IV. The basic idea is to assign the channel to the user with maximum signal to noise ratio (SNR), until the required number of channels for that user is satisfied. If some user k dominates too many channels, he simply chooses the L_k best channels and returns the rest to the channel pool for others.

TABLE IV
ALGORITHM FOR CHANNEL ALLOCATION UNDER CONSTANT-POWER WATER FILLING

Parameters: a set $\mathcal{I}$ of L channels available for allocation:
$\mathcal{I} = \{1, 2, \cdots L\}$, $|\mathcal{I}| = L$;
a set $\mathcal{K}$ of K users: $\mathcal{K} = \{1, 2, \cdots K\}$;
user k will be assigned L_k channels; $\sum_{k=1}^{K} L_k \leq L$.
a set $\mathcal{C}_k$ indicating the channels that are assigned to user k; initially, $\mathcal{C}_k = \emptyset$, $\forall k$;

Algorithm:

1. For any $i \in \mathcal{I}$,
   $i \rightarrow \mathcal{C}_k$ if $k = \arg\max_{k' \in \mathcal{K}} (\frac{P_{k'}\gamma_{k'i}}{L_{k'}})$.
2. For any user k with $|\mathcal{C}_k| = L_k$, allocation for user k is done, and then $\mathcal{K} = \mathcal{K} - \{k\}$, and $\mathcal{I} = \mathcal{I} - \mathcal{C}_k$.
   Otherwise, for those with $|\mathcal{C}_k| > L_k$,
   split $\mathcal{C}_k$ into $\mathcal{C}_k^1 \bigcup \mathcal{C}_k^2$, such that $|\mathcal{C}_k^1| = L_k$ and $\gamma_{kj} < \min_{i \in \mathcal{C}_k^1} \gamma_{ki}$, $\forall j \in \mathcal{C}_k^2$.
   $\mathcal{C}_k = \mathcal{C}_k^1$, $\mathcal{K} = \mathcal{K} - \{k\}$, and $\mathcal{I} = \mathcal{I} - \mathcal{C}_k$.
3. Stop if $\mathcal{K} = \emptyset$ or $\mathcal{I} = \emptyset$.
   Otherwise repeat from Step 1.

V. SIMULATIONS

We show how the algorithms work by the use of simple examples. For load balancing, consider a scenario where the spectrum is divided into $b = 10$ bands. The first band consists of 1000 channels, and each successive band has 200 more channels than the previous band. There are $N_u = 100$ users requesting channel resources. The requests are some integers (number of channels) uniformly distributed between 20 and 50. The initial load factor $\boldsymbol{\eta}$ has each of its element randomly selected from interval $[0 \;\; 1]$.

Fig. 2 shows the results from a simulation snapshot. It compares the load balancing performance of two proposed algorithms, i.e., the multi-level subset sum algorithm and the greedy algorithm. It can be seen that the bands with heavy traffic are not assigned any traffic under both algorithms. For the bands with assigned traffic, their final load factors are close to each other, showing that algorithms are working properly. The greedy algorithm has surprisingly good performance which is very close to that of the multi-level subset sum algorithm. By calculating the variance of η on assigned bands, we can see that for the multi-level subset sum algorithm, the value is

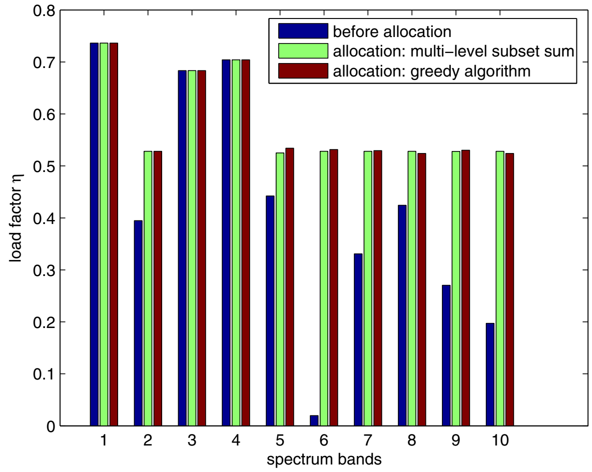

Fig. 2. Comparison of load balancing performance using the multi-level subset sum allocation and the greedy algorithm.

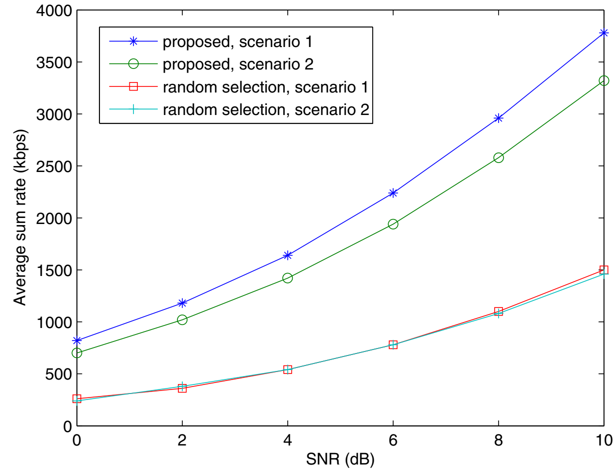

Fig. 3. Sum rate performance using proposed algorithm and random selection. Results are averaged from 100 simulation runs.

1.27×10^{-6}, and for the greedy algorithm it is 1.43×10^{-5}. Both are considerably small.

After load balancing, channel and power allocation is performed for each band. In the simulation, we assume that there are $K = 5$ users requesting $[10\ 20\ 30\ 20\ 15]$ channels, respectively. Each channel has a bandwidth of 20kHz. For simplicity, we assume users are of equal transmit power. The performance of the proposed algorithm is compared with that of the random selection algorithm where channels are randomly assigned to users. Results of two selected scenarios are presented in Fig. 3. In both scenarios, the available channels in the band are more than the total users' request, i.e., $L > \sum_{k=1}^{K} L_k$. The difference of two scenarios is that in case 1, $L - \sum_{k=1}^{K} L_k = 50$, and in case 2, $L - \sum_{k=1}^{K} L_k = 10$. It can be seen that the proposed algorithm has much better performance than random selection. When there are more frequency diversities as in scenario 1, the performance of the proposed algorithm is more superior, while random selection gains no benefit from diversity, as expected.

VI. Conclusions

In this paper, we consider resource allocation for cognitive radio networks. The spectrum is divided into multiple bands, and load balancing is performed to associate users of different requests with each band. Two algorithms of different complexity, namely the multi-level subset sum algorithm and the greedy algorithm, are proposed to achieve load balancing. Channel and power allocation is then performed in each band to maximize the sum data rate. Algorithm with constant-power water filling is proposed for channel/power allocation. Simulation examples are presented to demonstrate the effectiveness of the proposed algorithms.

References

[1] Federal Communications Commission (FCC), "Spectrum policy task force," Tech. Rep., ET Docket no. 02-135, Nov. 2002.

[2] I.F. Akyildiz, W.Y. Lee, M.C. Vuran, and S. Mohanty, "NeXt generation/dynamic spectrum access/cognitive radio wireless networks: a survey," *Computer Networks*, vol. 50, no. 13, pp. 2127–2159, 2006.

[3] S. Haykin, "Cognitive radio: brain-empowered wireless communications," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 2, pp. 201–220, 2005.

[4] F.F. Digham, "Joint power and channel allocation for cognitive radios," *IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 882–887, 2008.

[5] F.S. Chu and K.C. Chen, "Radio resource allocation in OFDMA cognitive radio systems," in *IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, Sept. 2007, pp. 1–5.

[6] A. Hoang and Y. Liang, "A two-phase channel and power allocation scheme for cognitive radio networks ," in *IEEE 17th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, 2006, pp. 1–5.

[7] D.I. Kim, L.B. Le, and E. Hossain, "Joint rate and power allocation for cognitive radios in dynamic spectrum access environment," *IEEE Transactions on Wireless Communications*, vol. 7, no. 12, pp. 5517–5527, 2008.

[8] J.W. Mwangoka and K.Z. Cao, "Robust End-to-End QoS Maintenance in Non-Contiguous OFDM Based Cognitive Radios," in *IEEE International Conference on Communications (ICC)*, 2008, pp. 2905–2909.

[9] S. Choudhary, S. Mishra, N. Desai, N.S. Priya, D. Chudasama, and R.V. Rajakumar, "A fair cognitive channel allocation method for cellular networks," in *Second International Workshop on Cognitive Radio and Advanced Spectrum Management (CogART)*, 2009, pp. 138–142.

[10] S. Martello and P. Toth, *Knapsack problems: algorithms and computer implementations*, John Wiley & Sons, 1990.

[11] T.H. Cormen, C.E. Leiserson, R.L. Rivest, and C. Stein, *Introduction to algorithms*, The MIT press, 2001.

[12] M. Dawande, J. Kalagnanam, P. Keskinocak, FS Salman, and R. Ravi, "Approximation algorithms for the multiple knapsack problem with assignment restrictions," *Journal of combinatorial optimization*, vol. 4, no. 2, pp. 171–186, 2000.

[13] C. Zeng, L.M.C. Hoo, and J.M. Cioffi, "Optimal water-filling algorithms for a Gaussian multiaccess channel with intersymbol interference," in *IEEE International Conference on Communications (ICC)*, 2001, vol. 8, pp. 2421–2427.

[14] W. Yu and J.M. Cioffi, "On constant power water-filling," in *IEEE International Conference on Communications (ICC)*, 2001, vol. 6, pp. 1665–1669.

[15] A. Giovanidis, T. Haustein, Y. Hadisusanto, A. Sezgin, and D. Kim, "Multiuser scheduling using equal power in allocated subcarriers for OFDM uplink," in *14th Asilomar Conference on Signals, Systems and Computers.*, 2006, pp. 1820–1824.