POWER ALLOCATION IN MULTI-CHANNEL COGNITIVE RADIO NETWORKS WITH CHANNEL ASSEMBLING

Lei Jiao[†], *Meisam Razaviyayn*[‡], *Enbin Song*[‡], *Zhi-Quan Luo*[‡], *and Frank Y. Li*[†]

[†] Dept. of Information and Communication Technology, University of Agder, Grimstad, Norway
[‡] Dept. of Electrical and Computer Engineering, University of Minnesota, Minneapolis, USA
Email: [†] {lei.jiao;frank.li}@uia.no, [‡] {razav002;luozq}@umn.edu, [‡] e.b.song@163.com

ABSTRACT

Consider power allocation for Secondary User (SU) packet transmissions on multiple channels with different channel conditions and variable Primary User (PU) arrival rates in a cognitive radio network. Two problems are studied in this paper. The first one is to minimize the collision probability with PUs and the second problem is to maximize the data rate while keeping the collision probability bounded. It is shown that the optimal solution for the first problem is to allocate all power onto the best channel based on a certain criterion. The second problem with per-channel power budget constraint is proved to be NP-hard and therefore a pseudo-polynomial time solution for the problem is proposed. When a total power budget for all channels is imposed in the second problem, a computationally efficient algorithm is introduced. The proposed algorithms are validated by numerical experiments.

1. INTRODUCTION

Spectrum access in Cognitive Radio Networks (CRNs) can be implemented in an Opportunistic Spectrum Access (OSA) manner [1], where SUs transmit over a band only if none of the PUs is transmitting in that band. By utilizing spectrum sensing, the SUs can decide to transmit if the sensing result indicates that all PU transmitters are inactive at this band.

In distributed CRNs with OSA approach, Medium Access Control (MAC) protocols usually work in a competing manner whereby the SUs compete for channel opportunities, with the winning SU using the available channels while other SUs have to wait for the next competition. When multiple available channels exist, channel assembling technique can be utilized by the winner in order to support higher data rate and further improve spectrum utility, as discussed in [2–5]. Traditionally, waterfilling is adopted for power allocation among multiple channels. However, this approach may lead to high probability of collision between SU and PU activities. When such collision happens, i.e., PUs appear during an SU packet transmission, SUs must release the channel immediately in order to make room for PUs, resulting a cost to SUs. More recently, the reference [6] introduced a risk-return model for SUs in which the cost of this collision in a given band is modeled as a rate loss depending on the power level allocated to this band. Under this model, the optimal power allocation strategy turned out to be similar to the traditional waterfilling. However, in practice, the full impact of such collision is much more than just the wasted transmission power or the associated rate loss. It includes other important ramifications, such as the resulting SU packet loss, the delay and the overhead in the handshake process between SU communication pairs. Modeling this collision just as a rate loss is insufficient.

In this work, we directly minimize or constrain the collision probability. Specifically, we consider two optimal power allocation problems for the case where SUs access the channels in a competing manner and only the winner can utilize the vacant channels for packet transmission after competition. One problem is to minimize the collision probability of an SU packet with PUs. The other is to maximize the capacity given the upper bound of SU packet collision probability.

The rest of the paper is organized as follows. The system model is given in Sec. 2 while the optimal power allocation problems are described and analyzed in Sec. 3. Then various algorithms are designed to solve the problems in Sec. 4. Numerical results and corresponding discussions are presented in Sec. 5, before the paper is concluded in Sec. 6.

2. SYSTEM MODEL

For notational convenience, we use an SU to indicate an SU communication pair in the following paragraphs. Assume there are M channels available to the winner after channel competition and sensing. Suppose a PU service requires only one channel and all of these channels have identical bandwidth B. Due to hardware constraint, an SU can assemble up to N channels for a packet transmission. Those channels can be either neighboring to each other or separated in the spectrum domain. Therefore, considering channel availability

This research is supported by the EU Seventh Framework Programme FP7-PEOPLE-IRSES under grant agreement no. 247083, project acronym S2EuNet. This research is carried out during a visit of the first author to the Dept. of Electrical and Computer Engineering, University of Minnesota.

and hardware constraint, the SU can utilize up to $\min\{M, N\}$ channels for a packet transmission.

When OFDM is utilized, each of those channels contains further S subchannels corresponding to the subcarriers in the system. The channel state, noise density and the SU's allocated power for the jth subchannels in channel i is denoted by $h_{i,j}$, $n_{i,j}$, and $p_{i,j}$ respectively, where $i \in I, I = \{1, \cdots, M\}$ and $j \in J$, $J = \{1, \cdots, S\}$. Each subcarrier has equal bandwidth b, where $Sb = B$. If a transmission scheme other than OFDM is performed where there are no subchannels, $h_{i,j}$, $n_{i,j}$, and $p_{i,j}$ will become h_i, n_i, and p_i.

Assume the arrival of the PU services follows Poisson process with rate λ_i in channel i, $i \in I$. In a period τ, the probability that there is no PU activity in channel i, denoted by $\mathcal{P}_i(\tau)$, is given by $\mathcal{P}_i(\tau) = e^{-\lambda_i \tau}$. Assume further that PU services are independent among different channels, the probability that there is no PU activity in a given channel set C_s during period τ, denoted by $\mathcal{P}_{C_s}(\tau)$, is

$$\mathcal{P}_{C_s}(\tau) = \prod_{i \in C_s} \mathcal{P}_i(\tau) = e^{-\sum_{i \in C_s} \lambda_i \tau}. \tag{1}$$

If there is no collision with the PUs, the time required to transmit a packet of the SU, denoted as T, is given by

$$T = \frac{L_p}{\sum_{i=1}^{M} \sum_{j=1}^{S} b \log(1 + |h_{i,j}|^2 p_{i,j}/(n_{i,j} b))}, \tag{2}$$

where L_p is the packet length and the denominator is the achieved capacity. Without loss of generality, we merge $n_{i,j} b$ and $|h_{i,j}|^2$ by defining $h'_{i,j} = |h_{i,j}|^2/(n_{i,j} b)$.

Let us define the channel usage indicator ξ_i, $i \in I$ as

$$\xi_i = \begin{cases} 1, & \sum_j p_{i,j} > 0, \\ 0, & \text{otherwise}, \end{cases} \tag{3}$$

where $\sum_i \xi_i \leq \min\{M, N\}$. This parameter indicates whether channel i is utilized by the SU or not.

We further assume that the set of assembled channels for the SU packet is fixed during its transmission. Based on Eqs. (1), (2), and (3), the probability that a packet is transmitted without collision with a PU activity can be formulated as

$$\mathcal{P}_r = \exp\left(-\frac{\sum_{i=1}^{M} \lambda_i \xi_i L_p}{\sum_{i=1}^{M} \sum_{j=1}^{S} b \log(1 + h'_{i,j} p_{i,j})}\right). \tag{4}$$

3. OPTIMIZATION PROBLEMS AND ANALYSIS

3.1. Minimizing the collision probability

Based on the above system model and for a given power budget, the optimization problem of minimizing the probability that an SU packet will collide with PUs, i.e., minimizing $1 - \mathcal{P}_r$, can be derived as

$$\min_{\{p_{ij}\}_{i \in I, j \in J}} \frac{\sum_{i=1}^{M} \lambda_i \xi_i L_p}{\sum_{i=1}^{M} \sum_{j=1}^{S} b \log(1 + h'_{i,j} p_{i,j})}, \tag{5}$$

$$\text{s.t. } \xi_i = \begin{cases} 1, & \sum_j p_{i,j} > 0, \\ 0, & \text{otherwise}, \end{cases}$$

$$1 \leq \sum_i \xi_i \leq \min\{M, N\},\ p_{i,j} \geq 0, \forall i, j,$$

$$\sum_i \sum_j p_{i,j} \leq p_t; \text{or } \sum_j p_{i,j} \leq p_t, \forall i \in I, \tag{6}$$

where p_t is the total power budget. As illustrated in (6), two cases for power constraint are considered, either there is a total power budget or there exists a power constraint for each channel. The condition $\sum_i \xi_i \geq 1$ is introduced so that at least one band is used by the winning SU to send its packet.

For a fixed set of selected channels and the packet length, the probability that an SU packet collides with PUs will be reduced if the data rate[1] increases. Since waterfilling is the optimal power allocation scheme for the total power budget case, once the channels are selected, waterfilling must be used. Similarly, in the per-channel budget constraint case, the maximum power should be utilized in each of the selected channels, while in subchannels within a particular channel the power is still allocated in the waterfilling manner.

Proposition 1 *The optimal solution for the problem (5) is to allocate the whole power to only one channel i which gives the minimum value of $\lambda_i L_p / \sum_{j=1}^{S} b \log(1 + h'_{i,j} p^*_{i,j})$, where $p^*_{i,j}$ is the solution of waterfilling for channel i with p_t.*

Proof We prove it by contradiction. Assume that $\sum_i \xi_i = \ell \geq 2$, i.e., ℓ channels are utilized as the optimal solution for transmission in the total power constraint case. Without loss of generality, we assume that those ℓ channels are sorted from low to high according to $\lambda_i L_p / \sum_{j=1}^{S} b \log(1 + h'_{i,j} p_{i,j})$, where $i \in \{1, \cdots, \ell\}$ and $\sum_i \sum_j p_{i,j} = p_t$. By dropping channel ℓ, i.e., setting $p_{\ell,j} = 0,\ \forall j$, we have

$$\frac{\sum_{i=1}^{\ell-1} \lambda_i L_p}{\sum_{i=1}^{\ell-1} \sum_{j=1}^{S} b \log(1 + h'_{i,j} p_{i,j})} \tag{7}$$

$$\leq \frac{\sum_{i=1}^{\ell} \lambda_i L_p}{\sum_{i=1}^{\ell} \sum_{j=1}^{S} b \log(1 + h'_{i,j} p_{i,j})}, \tag{8}$$

which is a contradiction since it gives us a better optimal point with smaller number of channels[2]. Similar result can be applied to the single channel power constraint case.

[1] The achieved data rate is determined by channel condition, power budget and coding/modulation scheme etc. Modern coding/modulation scheme can achieve data rate close to the Shannon capacity. In this work, we use data rate and capacity interchangeably.

[2] Note that if we do waterfilling again in the new set after dropping that channel, the denominator of Eq. (7) will increase since the portion of the power used for the channel that we dropped can be reused for the remaining channels. Therefore the inequality Eq. (7) becomes strict in this case.

3.2. Maximizing data rate with collision probability constraint

More generally, one would maximize the data rate while keeping the collision probability below a threshold value. Then the optimization problem becomes

$$\max_{\{p_{i,j}\}_{i\in I, j\in J}} \sum_{i=1}^{M}\sum_{j=1}^{S} b\log(1+h^{'}_{i,j}p_{i,j}), \tag{9}$$

$$\text{s.t.} \ \frac{\sum_{i=1}^{M}\lambda_i\xi_i}{\sum_{i=1}^{M}\sum_{j=1}^{S} b\log(1+h^{'}_{i,j}p_{i,j})} \le \gamma_0, \tag{10}$$

$$\xi_i = \begin{cases} 1, & \sum_j p_{i,j} > 0, \\ 0, & \text{otherwise}, \end{cases}$$

$$\sum_i \xi_i \le \min\{M,N\},\ p_{i,j} \ge 0, \tag{11}$$

$$\sum_i\sum_j p_{i,j} \le p_t; \text{or} \sum_j p_{i,j} \le p_t, \forall i \in I,$$

where $\gamma_0 = -\log(1-\mathcal{P}_{rc0})/L_p$ and $\mathcal{P}_{rc0}$ is the maximum tolerable level of the collision probability.

If we ignore the hardware constraint in (11) and consider only per-channel power constraint, the problem becomes

$$\max_{\{p_i\}_{i\in I}} \sum_{i=1}^{M}\sum_{j=1}^{S} b\log(1+h^{'}_{i,j}p_{i,j}), \tag{12}$$

$$\text{s.t.} \ \frac{\sum_{i=1}^{M}\lambda_i\xi_i}{\sum_{i=1}^{M}\sum_{j=1}^{S} b\log(1+h^{'}_{i,j}p_{i,j})} \le \gamma_0,$$

$$\xi_i = \begin{cases} 1, & \sum_j p_{i,j} > 0, \\ 0, & \text{otherwise}, \end{cases}$$

$$\sum_{j=1}^{S} p_{i,j} \le p_t,\ p_{i,j} \ge 0. \tag{13}$$

Proposition 2 *The optimization problem* (12)-(13) *which is a special case of the optimization problem* (9)-(11) *is NP-hard.*

Proof Let $p^{'}_{i,j}$ be the solution of (12)-(13) and let us also define $q_i = \sum_{j=1}^{S} p^{'}_{i,j}/p_t$. Since $\sum_{j=1}^{S} p^{'}_{i,j}$ is either zero or p_t, $q_i \in \{0,1\}$. Let $v_i = \sum_{j=1}^{S} b\log(1+h^{'}_{i,j}p^{*}_{i,j})$, where $p^{*}_{i,j}$ denotes the waterfilling solution in channel i with power budget p_t. Thus, the problem becomes

$$\max_{\{q_i\}_{i\in I}} \sum_i v_i q_i, \tag{14}$$

$$\text{s.t.} \ \sum_i \lambda_i q_i / \sum_i v_i q_i \le \gamma_0,\ q_i \in \{0,1\}.$$

Given $\lambda_i - \gamma_0 v_i \le 0$ for a specific channel i, we must set $q_i = 1$, because this choice of variable satisfies the constraint and increases the value of objective function. On the other hand, for the channels that $\lambda_i - \gamma_0 v_i > 0$, we must solve the following optimization problem:

$$\max_{\{q_i\}_{i\in I'}} D + \sum_i v_i q_i, \tag{15}$$

$$\text{s.t.} \ \sum_i (\lambda_i - \gamma_0 v_i) q_i \le C,\ q_i \in \{0,1\},$$

where $I^{'} = \{i|\lambda_i - \gamma_0 v_i > 0, i \in I\}$, $C = -\sum_{j\in I''}(\lambda_j - \gamma_0)v_j$, $D = \sum_{j\in I''} v_j q_j$ and $I'' = I - I'$ is the complement of the set I'. Clearly, (15) is a knapsack problem. Moreover, we can start from an instance of a knapsack problem and build the equivalent power allocation problem (12)-(13).

4. ALGORITHMS FOR POWER ALLOCATION

In what follows, we suggest different algorithms for the data rate maximization problem under various power constraints.

4.1. Power allocation with per-channel power constraint

For the per-channel power constraint case, based on our previous discussions, we can re-formulate the problem as

$$\max_{\{q_i\}_{i\in I}} \sum_i v_i q_i, \tag{16}$$

$$\text{s.t.} \ \sum_i w_i q_i \le 0,\ \sum_i q_i \le \min\{M,N\},\ q_i \in \{0,1\},$$

where $w_i = \lambda_i - \gamma_0 v_i$.

Inspired by the dynamic programming algorithm for the knapsack problem, we propose a pseudo-polynomial time algorithm as follows. Define $m(i,x,n)$ to be the maximum value of the objective function that can be attained with weight less than or equal to x, by choosing channels (or *items* in the knapsack problem) from the set $\{1,2,\ldots,i\}$ and choosing at most n channels. It is easy to see that the following equations hold:

$$m(i,x,0) = \begin{cases} 0; & x \ge 0, \\ \text{infeasible}; & \text{otherwise}, \end{cases}$$

$$m(0,x,n) = \begin{cases} 0; & x \ge 0, \\ \text{infeasible}; & \text{otherwise}, \end{cases}$$

$$m(1,x,n) = \begin{cases} v_1; & n \ge 1, x \ge w_1, \\ 0; & n = 0, x \ge 0, \\ \text{infeasible}; & x < \min\{0, w_1\}, \end{cases}$$

$$m(i,x,n) = \begin{cases} \max\{A, B+v_i\}; & \text{both A and B feasible}, \\ A; & \text{A feasible, B infeasible}, \\ B+v_i; & \text{B feasible, A infeasible}, \\ \text{infeasible}; & \text{both A and B infeasible}, \end{cases}$$

where $A = m(i-1,x,n)$ and $B = m(i-1,x-w_i,n-1)$.

Since neither w_i nor v_i are required to be integers, a top-down approach in dynamic programming is utilized. Therefore, the final result, i.e., $m(M,0,\min\{M,N\})$, can be calculated in a recursive manner through dynamic programming.

4.2. Power allocation with total power constraint

We now introduce a highly efficient heuristic for the total power constraint case as illustrated in Algorithm 1. This algorithm is based on the fact that a channel with a smaller $\lambda_i L_p / \sum_{j=1}^{S} b\log(1+h^{'}_{i,j}p_{i,j})$, $\forall i \in I$ and $\sum_j p_{i,j} = p_t$, may better satisfy the probability constraint.

Define $[R, \mathbf{p}] := wf(m, n, p)$ as the waterfilling function using from the m-th to the n-th channels with power budget p, where R is the resulted capacity and $\mathbf{p}$ is the resulted power allocation vector. In this algorithm, firstly waterfilling is done for each of channel individually with the total power budget constraint. By doing so, we can check the feasibility of the problem and sort the channels from low to high according to $\lambda_i L_p / \sum_{j=1}^{S} b \log(1 + h'_{i,j} p_{i,j})$, $\forall i \in I$ and $\sum_j p_{i,j} = p_t$. Let this new ordered channel set be I_o. Based on the resulted ranking, we form a set with channel index from the first one to the largest possible one, i.e., make the set have as many channels as possible while keeping the probability and the hardware constraints satisfied. The reason is that with a total power budget, the larger number of channels we utilize, the higher the capacity it can potentially achieve through waterfilling.

Algorithm 1 : A sub-optimal algorithm

for $i := 1$ to M **do**
  $[R_i, \mathbf{p}] := wf(i, i, p_t)$.
end for
if $\forall\ \lambda_i L_p / R_i > \gamma_0$ **then**
  Problem infeasible.
else
  Rank channels according to $\lambda_i L_p / R_i$ from low to high.
  if $N \geq M$ **then**
    Return $[R, \mathbf{p}] := Search(M)$.
  else
    $[Capa, \mathbf{p}'] := Search(N)$.
    if $Search(N) = wf(1, N, p_t)$ **then**
      for $i := N + 1$ to M **do**
        $[R, \mathbf{p}] := wf(i - N + 1, i, p_t)$.
        if the solution is feasible and $R > Capa$ **then**
          $Capa := R$ and $\mathbf{p}' := \mathbf{p}$.
        end if
      end for
    end if
    Return $[Capa, \mathbf{p}']$.
  end if
end if

In Algorithm 1, there is a function $[R, \mathbf{p}] := Search(s)$ which is explicitly given in Algorithm 2. This function performs based on the bisection method. The variable s in this function indicates the searching range, i.e., from the first to the sth channel in the new ordered channel set I_o. The returned values $[R, \mathbf{p}]$ are based on the largest feasible subset with elements starting from the first channel consecutively, up to the sth one in the new ordered channel set. The function $[R, \mathbf{p}] := Search(s)$ can always find a feasible solution if it is called, since the feasibility of the problem has been checked and the channels are ranked accordingly in Algorithm 1.

Algorithm 2 : $Search(s)$

Let $m := 0$, $n := 1$, $f := 1$, $capa := 0$, $\mathbf{p}' := 0$.
repeat
  $m := m + \lceil (1/2)^n s \rfloor f$, $[R, \mathbf{p}] := wf(1, m, p_t)$.
  if the solution is feasible **then**
    $f := 1$.
    if $R > capa$ **then**
      $capa := R$ and $\mathbf{p}' := \mathbf{p}$.
    end if
  else
    $f := -1$.
  end if
  $n := n + 1$.
until $\lceil (1/2)^n s \rfloor = 0$. Return $[capa, \mathbf{p}']$.

5. NUMERICAL RESULTS AND DISCUSSIONS

In this section, the performance of the proposed algorithms are evaluated via numerical experiments. In both of the per-channel and the total power budget constraint cases, two scenarios when $N \geq M$ and $N < M$ are investigated. The default parameters are summarized in Table 1. In order to evaluate the performance of the proposed algorithms, an exhaustive search algorithm is considered as the benchmark. All the illustrated results are the average values of over 100 runs.

Table 1. Parameters for performance analysis.

Notation	Values
Number of subchannels (S)	8
Channel state ($h_{i,j}$)	Rayleigh distributed with parameter 1/0.6552
Noise density ($n_{i,j}$)	10^{-10} W/Hz
Power budget (p_t)	8×10^{-3} W
Channel bandwidth (B)	2×10^{6} Hz
PU Poisson arrival rate (λ_i)	Uniformly distributed between 40 to 100 times/s
Packet length (L_p)	8000 bit
Collision probability ($\mathcal{P}_{rc0}$)	3%

5.1. Per-channel power constraint case

The proposed pseudo-polynomial time algorithm is compared with the exhaustive search algorithm in two aspects: The data rate achieved and the computational complexity as represented by the machine running time. In our numerical experiments we observed that the pseudo-polynomial time algorithm always finds the optimal solution, therefore we do not plot the results explicitly. The running time with respect to the number of channels M is plotted in Fig. 1 when $N \geq M$, i.e., with sufficient hardware.

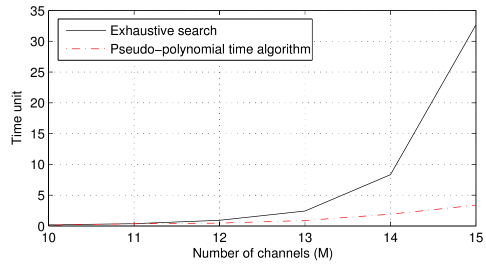

Fig. 1. Time consumption as a function of M when $N \geq M$.

As observed from Fig. 1, when the number of total channels grows, the time used by exhaustive search increases dramatically. We have also observed that the pseudo-polynomial time algorithm consumes slightly more time than the exhaustive search does when M is small, i.e, $M < 7$ in this example, although not observable in the current plotting. It means that when only a few channels are available, the exhaustive search method is a good option. However for large M, the pseudo-polynomial time algorithm through dynamic programming is preferable. Similar results have been observed when $N < M$ however not illustrated here due to page limit.

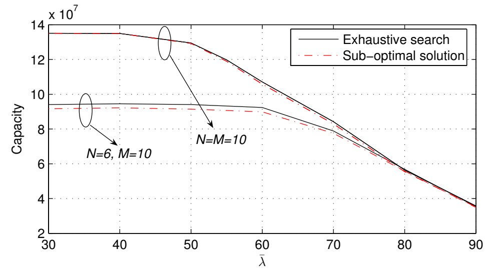

Fig. 2. Capacity as a function of PU arrival rate.

5.2. Total power constraint case

In Fig. 2, we illustrate the capacity as a function of the mean value of the PU arrival rate among channels in the total power budget case. The total power budget is 8×10^{-2} W, and the PU Poisson arrival rate among different channels, λ_i, is uniformly distributed with the mean value $\bar{\lambda}$ and the variance of 300, while other parameters follow the default values.

Two cases, $N = M = 10$, and $N = 6$ and $M = 10$, are studied. From Fig. 2, we can observe that the capacity of the algorithms in both cases is relatively stable initially and decreases as the average PU arrival rate increases. When the mean arrival rate of PU service is small, most of the channels can be utilized for packet transmission while keeping the probability constraint satisfied. When the mean PU arrival rate becomes larger, the number of channels that can make the probability constraint satisfy decreases. Given the same total power budget constraint, with smaller number of assembled channels, i.e., less bandwidth, the capacity will be reduced. Comparing the capacity of the sub-optimal and the exhaustive search algorithms, the capacity of sub-optimal algorithm is quite close to the result of the exhaustive search method.

Furthermore, with respect to computational complexity, the number of times for executing the waterfilling algorithm is only proportional to M using the sub-optimal algorithm while it is exponential to M in the exhaustive search method.

6. CONCLUSIONS

In this paper, power allocation in CRNs is considered from two aspects, minimizing collision probability with PUs and maximizing the capacity with constraint collision probability. The optimal solution of the first problem is provably to put full energy in the single best channel while the second problem is proved to be NP-hard in the per-channel power constraint case. Therefore a dynamic programming method is proposed for power allocation with per-channel power constraint. A highly efficient heuristic is introduced for power allocation with total power constraint. As expected, the numerical results demonstrate that dynamic programming achieves the optimized result, and that the heuristic algorithm is capable of achieving data rates close to the global optimal at very low computational complexity.

7. REFERENCES

[1] R. Zhang, Y. Liang, and S. Cui, "Dynamic Resource Allocation in Cognitive Radio Networks: A Convex Optimization Perspective," *IEEE S. P. Mag.*, vol. 27, no. 3, pp. 102-114, May 2010.

[2] H. Su and X. Zhang, "Cross-Layer Based Opportunistic MAC Protocols for QoS Provisionings over Cognitive Radio Mobile Wireless Networks," *IEEE JSAC*, vol. 26, no. 1, pp. 118-129, Jan. 2008.

[3] J. Jia, Q. Zhang, and X. Shen, "HC-MAC: A Hardware-Constrained Cognitive MAC for Efficient Spectrum Management," *IEEE JSAC,* vol. 26, no. 1, pp. 106-117, Jan. 2008.

[4] H. A. B. Salameh, M. M. Krunz, and O. Younis, "MAC Protocol for Opportunistic Cognitive Radio Networks with Soft Guarantees," *IEEE Trans. Mobile Computing,* vol. 8, no. 10, pp. 1339 - 1352, Oct. 2009.

[5] A. C. Hsu, D. S. L. Wei, and C.-C. J. Kuo, "A Cognitive MAC Protocol using Statistical Channel Allocation for Wireless Ad-hoc Networks," *Proc. IEEE WCNC,* Hong Kong, Mar. 2007.

[6] Z. Hasan, E. Hossain, C. Despins, and V. K. Bhargava, "Power Allocation for Cognitive Radios Based on Primary User Activity in an OFDM System", *Proc. IEEE Globecom*, New Orleans, USA, Nov. 2008.