Cooperative-Parallel Spectrum Sensing in Cognitive Radio Networks Using Bipartite Matching

Behzad Shahrasbi and Nazanin Rahnavard
School of Electrical and Computer Engineering Oklahoma State University, Stillwater, OK 74078
Emails: {behzad.shahrasbi, nazanin.rahnavard}@okstate.edu

***Abstract*—Spectrum sensing is one of the key steps for implementing the cognitive radio-based systems. The efficiency and the effectiveness of spectrum sensing have a profound impact on the performance of the cognitive users. In this paper, we propose two cooperative-parallel spectrum sensing algorithms. The cooperation greatly reduces the sampling time for each secondary user and increases the efficiency. Our proposed algorithms utilize adaptive schemes as well as the graph theoretical analysis to obtain the best strategy for channel sensing in the secondary users. In this work, we model the cooperative spectrum sensing problem with a bipartite graph. Assigning channel sensing tasks to the secondary users corresponds to finding the perfect matching on that graph. Two different algorithms are developed based on the different complexity levels of the underlying matching algorithms. The performances of these algorithms are compared with each other and with other related algorithms from the literature.**

I. Introduction

Cognitive radio (CR) is a promising solution to alleviate today's spectrum deficiency [1], which is caused by an increased demand for wireless technologies. Current fixed spectrum allocation strategies under-utilize this valuable resource. Therefore, CR was proposed to mitigate the under-utilization of the spectrum and to make spectrum allocation more efficient [1, 2]. According to the CR paradigm, in addition to the existing licensed users of the spectrum, a new type of users is defined who are allowed to access the spectrum given that they do not interfere with the licensed users. The licensed users are also called primary users (PUs) and the unlicensed users whose access to the resources is *opportunistically* possible are called secondary users (SUs) or cognitive users. The under-utilized parts of the spectrum are called spectrum holes [2]. The ideal CR is able to efficiently detect and utilize these holes.

Spectrum sensing is the key CR task which is responsible for finding the spectrum holes. Finding more spectrum holes means more opportunities for SUs to transmit. Several detection methods are used for spectrum sensing in the literature such as energy detector [3–5], waveform-based detector [5], cyclostationarity-based sensing [5], and matched-filter based detector [5]. In this work, we consider *energy detector* due to its simplicity and its ability to detect with the least amount of *a priori* knowledge about PUs signals. The IEEE standard for CRs is *IEEE 802.22* [6], in which the spectrum sensing is carried out in two consecutive stages in which the first stage is an energy detector.

The cooperation among the SUs are also considered in many papers to mitigate the problems such as hidden terminal, shadowing, and fading. Moreover, it has been shown that cooperation decreases the sensing time [7, 8]. In these works, the cooperation is used to enhance the sensing of one specific channel. In other words, all SUs are assigned to sense the same PU channel. Cooperation can be also implemented in another level in which each SU senses a unique channel which is different than the channel that other SUs sense [9, 10]. In [9] such cooperation is introduced and it is called *parallel* cooperative spectrum sensing. In this work, we also consider that different SUs sense different channels simultaneously.

The challenges that are imposed by the cooperation are an increased complexity and the necessity of a common control channel. The common control channel can be implemented [11] via a dedicated band, unlicensed band, or underlay (i.e. using the same band with much lower energy such that it does not create interference). The control channel can be also incorporated in the same transmission and sensing channel using TDMA schemes [12].

The limitations of RF frontend has required the channel sensor to be able to sense one channel at a time [13, 14]. Therefore, it is more efficient if an SU senses a channel which has a high probability of being empty as opposed to sensing a random channel. Accordingly, the estimation of PUs arrival model have been considered in some recent studies [4, 15]. For example, an adaptive channel sensing strategy is proposed in [15]. The authors have shown that using the previous history of the channels, an SU chooses to sense the best channel it believes would provide the most reward (highest probability of being empty). Additionally in [16] an efficient sensing scheme based on optimizing the sensing period and sensing sequence using PU's channel occupancy, is proposed.

The contribution of this paper is to propose two cooperative-parallel spectrum sensing algorithms that are highly efficient in terms of sensing requirements. Among all available PU channels, each SU senses only one PU channel. This PU channel is uniquely determined for each SU in a central node (CN) that has all the sensing information from the previous sensing durations. We also minimize the overhead of transmissions over the common control channel.

The remainder of this paper is organized as follows. In Section II, we introduce our model and assumptions for the PU and SU channels. In Section III, first, we present our ideas for relating a spectrum sensing problem to a similar problem in graph theory. Then, using the mathematical structure in graph theory, we propose two algorithms for implementing the cooperative-parallel spectrum sensing. In Section IV, the experimental results of implementing the proposed algorithms are depicted. Finally, Section V provides the discussion of the results of this work and concludes the paper.

II. System Model

The problem that we consider in this paper can be represented with a toy example in Figure 1. In this Figure, PUs occasionally use their licensed channels for transmission of their own traffic. The existing SUs are trying to find the opportunities that the PUs are not available thus they can step in and use those channels. In this framework, SUs are equipped with frontends that can only sense one channel at a time and no control channel exists besides *CH 1, CH 2, and CH 3* (i.e. PU channels). We model a PU's activity in time

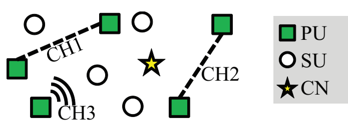

Fig. 1. Centralized cooperative spectrum sensing

by a Markov model (MM) with the state transition matrix $A = \left[\begin{smallmatrix} a_{00} & a_{01} \\ a_{10} & a_{11} \end{smallmatrix}\right]$ as shown in Figure 2. According to this model, each PU channel can be in one of the two states busy (on) and empty (off), which represent whether or not PU is active in that time slot. The probabilities a_{00}, a_{01}, a_{10}, and a_{11} are the transition probabilities which represent the probability of staying in the busy state, transition from busy state to empty state, transition from empty state to busy state, and staying in the empty state, respectively. The transition might happen in between two consecutive sensing attempts of that channel. We assume all PUs have the same transition probabilities.

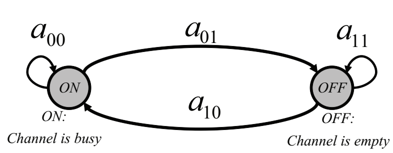

Fig. 2. Model of primary user arrival

Assuming the TDMA sharing policy for the sensing and transmission, we define the following time durations:

- Sampling period (T): The amount of time between two consecutive sensing of an SU.
- Sensing duration (T_S): The amount of time SU requires to sense a channel.
- Transmission duration (T_X): The amount of time SU obtained to send its own traffic in a sampling period.

These sensing and transmission durations for a SU are represented in Figure 3 and we have $T = T_S + T_X$. SUs periodically choose a channel to sense and send sensing results to CN. Then, CN coordinates the channel sensing and channel access at SUs. Based on the gathered information and previous channel occupation history, CN allocates the channel that each SU has to sense in the subsequent sensing period.

Sampling period (T) should be short enough to avoid PU quality of service (QoS) degradation. On the other hand, more transmission opportunities can be obtained from the channel as the transmission time increases. The sensing efficiency factor (η_{sense}) can be defined as

$$\eta_{sense} = \frac{T_X}{T} = \frac{T_X}{T_X + T_S}. \tag{1}$$

In CR implementation such as IEEE 802.22, T has a fixed and pre-defined value. Therefore, in (1), the only factor that can be adjusted to increase the efficiency is the sensing duration (T_S). The typical values for T and T_S are $200ms$ and $5ms$ [6, 16], respectively. In Section III-A, we show T_S can be reduced using the parallel cooperation and history of the channels. We assume n SU exist, and they are waiting for the opportunity to transmit. Moreover, m PU channels exist and the channel occupation follows the *on-off* model of Figure 2. In the toy example of Figure 1, $n = 4$ and $m = 3$.

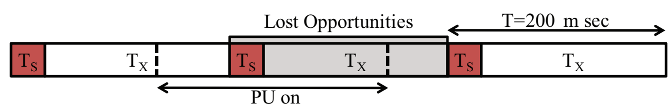

Fig. 3. Channel sensing cycles at SUs.

Since we employ energy detectors in SUs, an extra constraint would be imposed on the system. To avoid interference from other SUs, the sensing cycles in all SUs should be synchronized. The synchronization is performed in CN.

III. Cooperative-Parallel Spectrum Sensing using Bipartite Matching and Adaptive Learning

In Section II, the cooperative-parallel spectrum sensing for CR has been introduced. In this section, we describe the algorithms to allocate the channels to SUs for sensing. CN receives information from the SUs based on their sensing experience and decides which channels will be sensed in the next sensing period by each SU. The messages that are communicated between each SU and CN are in form of *vectors of probability distributions*. In other words, every SU keeps the history of previous PU channel sensing in a vector with length m. Let $\overline{x}_i = [x_{i,1}, \ldots, x_{i,m}]$ be the probability vector of the i^{th} SU and we use the approach in [15] to find and update $x_{i,j}$ at each T_S. Every element of this vector ($x_{i,j}$) is the probability that the i^{th} SU chooses the j^{th} PU channel to sense in the next sensing cycle. Each SU calculates this probability iteratively based on the new channel sensing results. Roughly speaking, every channel sensing that finds an empty channel increases the corresponding probability for that channel and every channel sensing that finds a busy channel decreases that probability. Our proposed approach is as follows. Each SU sends its probability vector to CN. After CN receives $\underline{x}_1, \ldots, \underline{x}_n$, it determines the best channel that each SU should sense. We propose to use bipartite matching from the graph theory for these optimal channel allocations.

We propose to use bipartite matching algorithms from the graph theory to allocate the available channels to the SUs. In the rest of this section, first, two cooperative-parallel spectrum sensing algorithms for CRs are introduced. Then, we formulate the problem of minimizing the graph size. Finally, we consider the effect of non-ideal channel sensing in the performance of the proposed algorithms.

A. Cooperative-Parallel Channel Sensing Algorithms

In this section, we show that the problem of cooperative-parallel spectrum sensing can be mapped into a bipartite matching problem. Accordingly if we can find a perfect matching over the corresponding bipartite graph then all PU channels would be sensed by SUs (i.e., all possible opportunities are found). In our proposed bipartite graph, we assignvertices of one side of the graph to SUs (i.e., n vertices) and vertices of the other side to the graph to PU channels (i.e., m vertices). To assign edges between these two sets of vertices we employ two different strategies as follows.

1) Creating a connected bipartite graph with minimal edges to have perfect matching (Algorithm 1).
2) Creating a weighted complete bipartite graph (Algorithm 2).

In the first strategy, we assign an edge between the vertex corresponding to i^{th} SU and the vertex corresponding to j^{th} channel if and only if $x_{i,j}$ is greater than a *threshold* δ. This threshold controls the number of edges in the graph. In Section III-B, we find this threshold through analysis. The greater the number of edges the more complex is the matching algorithm. On the other hand, if the threshold is too high, some PU channels may not be assigned for sensing and the opportunities are lost. Therefore, the optimal value for the threshold can be found by minimizing the number of edges given the graph is abiding the condition of Hall's theorem [17]. Equivalently, this condition implies that the corresponding bipartite graph be an expander graph with expansion factor greater than 1 (i.e. $\alpha > 1$) [18].

In second strategy, the graph is fully connected. In other words, all edges between the two sets of the vertices are connected. It is possible to inversely relate the probability of finding a transmission opportunity by the weight of each edge that connects the corresponding SU to that channel. Therefore, the higher the probability, the smaller the weight of that node would be. Using this allocation strategy, the complete graph that we obtain always satisfies the Hall's theorem condition if $m \geq n$. Yet, it is more complex to find the minimum weight allocation in this case [17]. The weights of each edge is obtained by $w_{i,j} = \frac{1}{x_{i,j}+\epsilon}$, where $w_{i,j}$ is the weight of the edge between the i^{th} SU and the j^{th} PU channel and ϵ is very small constant to avoid unbounded weights.

Employing any of the two edge allocating strategies generates different bipartite graphs that require different matching algorithms. In the first case, a simple augmenting path algorithm [17] can solve the problem yet, the second problem requires application of more complex methods such as Hungarian algorithm [17]. The following algorithms are proposed to perform cooperative sampling based on the choice of the edge allocation strategies. In the following algorithms $N(i)$ represents the set of neighboring vertices of vertex i and δ represents the threshold probability for connecting an edge.

Algorithm 1 finds the perfect matching by thresholding the edge connections. Therefore, the underlying bipartite graph becomes sparse i.e. only a small portion of edges are connected in the graph. The computational complexity of the matching algorithm which is carried out in CN is $O(mn)$ [19].

Algorithm 1: Channel allocation for efficient sensing with thresholding.

```
Initializing:
for i = 1 to n do
    Randomly select a PU channel to sense.
    x̄_i is updated based on Alg1 [15].
    for i = 1 to n do
        x̄_i is transmitted to CN.
    end for
    CN finds the threshold δ and transmit it to SUs.
end for
while SUs are interested in an empty channel do
    for i = 1 to n do
        for j = 1 to m do
            if x_{i,j} > δ then
                An edge is connected between vertex i and
                vertex j.
                The corresponding i, j which determine the edge
                uniquely are transmitted to CN.
            end if
        end for
    end for
    Run the augmenting path algorithm [17] in CN.
    The obtained matching results are transmitted to SUs.
    for i = 1 to n do
        The allocated channel is sensed.
        x̄_i is updated based on Alg1 [15].
    end for
end while
```

Algorithm 2 finds the perfect matching based on the weighted bipartite matching. The computational complexity of the weighted matching algorithm (Hungarian algorithm) is $O(mn^2)$ [19].

B. Obtaining the Optimal Graph Size in Algorithm 1

In this section, we provide the analysis for finding the threshold δ for connecting an edge between any two vertices of the bipartite graph. The higher the threshold the fewer the number of edges in the bipartite graph and SUs will be more selective in choosing a channel to sense. On the other hand, the threshold should be low enough to ensure that a perfect matching can be found on the graph. The following theorem relates δ with the probability of finding the perfect matching.

Theorem 1: For a bipartite graph $A = (N, E)$ with $N = X \bigcup Y$, $|X| = n$ and $|Y| = m$, and $n \leq m$ ($|.|$ represents the

Algorithm 2: Channel allocation for efficient sensing with weighted matching.

Initializing:
for $i = 1$ to n **do**
  Randomly select a PU channel to sense.
  $\overline{x}_i$ is updated based on *Alg1* [15].
end for
while SUs are interested in an empty channel **do**
  for $i = 1$ to n **do**
    $\overline{x}_i$ is transmitted to CN.
  end for
  Weights are updated in CN according to $w_{i,j} = \frac{1}{x_{i,j}+\epsilon}$.
  Run the *Hungarian* algorithm [17] in CN.
  The obtained matching results are transmitted to SUs.
  for $i = 1$ to n **do**
    The allocated channel is sensed.
    $\overline{x}_i$ is updated based on *Alg1* [15].
  end for
end while

cardinality of the set), a perfect matching $M \subseteq E$ from X to Y exists with probability $1 - p_f(\delta)$. Here δ is the threshold on the probability of existing an edge between any two nodes from sets X, Y and $p_f(\delta)$ is given as follows.

$$p_f = \sum_{i=1}^{n}\sum_{j=1}^{i}\left(\frac{\binom{n}{i}}{2^n}\binom{n}{d_{ij}}p(\delta)^{d_{ij}}(1-p(\delta))^{n-d_{ij}}\right)^j \sum_{d_i^m=1}^{i} i(I_{1-p(\delta)}(n-d_i^m, d_i^m+1)^{i-1}\binom{i}{d_i^m}p(\delta)^{d_i^m}(1-p(\delta))^{i-d_i^m} \frac{\sum_{k=d_i^m}^{i-1}\binom{m}{k}[\prod_{l=1}^{i}\binom{k}{d_{il}} - \binom{k}{k-1}\prod_{l=1}^{i}\binom{k-1}{d_{il}}]}{\prod_{l=1}^{i}\binom{m}{d_{il}}}, \tag{2}$$

where d_{ij} represents the degree of j^{th} vertex in any subset of X with the cardinality i and d_i^m is the maximum of d_{ij} over all possible values of j and $I_x(.,.)$ represents regularized incomplete beta function.

Proof: To verify (2), first, we find the probability of failure an specific subset of X with known degrees is assumed. The cardinality of this subset is $S = a$ and the degrees of the vertices in the set are $\underline{D} = \underline{d}_a = [d_{a1}, d_{a2}, \ldots, d_{aa}]$. In general, S is a random variable that represents the cardinality of any subset of X, $\underline{D}$ is a random vector that represents the degree of each node in any subset of X. The Hall's theorem states that a perfect matching can be found on a bipartite graph if and only if for any subset of X with cardinality of $S = a$, $N(S = a, \underline{D} = \underline{d}_a) \geq a$. Where $N(S = a, \underline{D} = \underline{d}_a)$ represents the set of neighboring nodes of any subset of X with node degrees $\underline{D} = \underline{d}_a$ and cardinality $S = a$. Therefore, given $\underline{d}_a$ and a the probability of not satisfying this condition is $p_f|_{S=a,\underline{D}=\underline{d}_a} = P\left(N\left(S = a, \underline{D} = \underline{d}_a\right)\right) < a$. Given the size and the degree of the nodes in any subset is known (i.e. a and $\underline{d}_a$, respectively),

$$p_f|_{S=a,\underline{D}=\underline{d}_a} = \frac{\text{\# of cases that Hall's theorem condition is not satisfied}}{\text{\# of all possible cases}} \tag{3}$$

The total number of possible ways for connecting vertices of a subset of X with cardinality a to the other set of vertices with cardinality m is equal to $\prod_{l=1}^{a}\binom{m}{d_{al}}$. On the other hand the number of ways that a subset of X with cardinality a has fewer neighbors than a in set Y is equal to,

$$\sum_{k=d_a^m}^{a-1}\binom{m}{k}[\prod_{l=1}^{a}\binom{k}{d_{al}} - \sum_{q=d_a^m}^{i-1}\binom{q}{i}\prod_{l=1}^{a}\binom{q}{d_{al}}].$$

Where $d_a^m = \max\{\underline{d}_a\}$. Therefore, equation (3) can be written as follows,

$$p_f|_{S=a,\underline{D}=\underline{d}_a} = \frac{\sum_{k=d_a^m}^{i-1}\binom{m}{k}[\prod_{l=1}^{a}\binom{k}{d_{al}} - \sum_{q=d_a^m}^{i-1}\binom{q}{i}\prod_{l=1}^{a}\binom{q}{d_{al}}]}{\prod_{l=1}^{a}\binom{m}{d_{al}}}. \tag{4}$$

d_a^m is also a random variable that its pmf function, $f_{(a)}(d_a^m)$, using the order statistics can be obtained as follows,

$$f_{(a)}(d_a^m) = a(I_{1-p(\delta)}(a - d_a^m, d_a^m + 1))^{a-1} \binom{a}{d_a^m}p(\delta)^{d_a^m}(1-p(\delta))^{a-d_a^m}, \tag{5}$$

where $p(\delta)$ is the probability of existing an edge between any two nodes and it is determined by the threshold δ. To find the total p_f it is necessary to average over S, $\underline{D}$, and d_a^m.

$$p_f = E_S\{E_{D|S=a}\{p_f|_{S=a,\underline{D}=\underline{d}_a}\}\} \tag{6}$$

The probability of having cardinality a for any subset of X is given by,

$$p(S = a) = \frac{\binom{n}{a}}{2^n}. \tag{7}$$

The probability that any subset of X having degree distribution $\underline{d}_a$ given the cardinality of that subset is a, is as follows,

$$p(\underline{D} = \underline{d}_a | S = a) = \prod_{i=1}^{a} p(D_i = d_i); \quad D_i \sim \text{BIN}(n, p(\delta)) \tag{8}$$

$p(\delta)$ is the outcome of random variable Π. To find the distribution of Π, we define random variable $\underline{Q}_i$ which its outcomes denote the probability vector of selecting vertices of Y at every vertex of X. We also assume the m vertices of Y are equally selected over a long period of time. Therefore, we find the probability distribution of $\underline{Q}_i$'s,

$$\underline{Q}_i \sim \text{Dir}([1/m, \ldots, 1/m]_{m\times 1}) \quad i = 1, \ldots, n. \tag{9}$$

where Dir(.) represents Dirichlet distribution. Therefore, the probability distribution of selecting a specific vertex of set Y by vertex i of set Π_i is obtained by calculating the marginal distribution of $\underline{Q}_i$ for that vertex (i.e. the distribution of the probability of existing an edge) which is,

$$\Pi_i \sim \beta\left(\frac{1}{m}, \frac{m-1}{m}\right). \tag{10}$$

Assuming vertices of set X are *iid*, the subscript i can be dropped. Therefore, $p(\delta)$ is obtain putting the probability of

$\Pi > \delta$ where Π is distributed based on (10). By substituting (6), (8), and (7) in (3), Equation (2) follows. Therefore, we can work backwards and set the threshold δ such that p_f remains less than any arbitrary small value. ■

C. Non-ideal Sensing

In the previous sections, we assumed that SUs have ideal sensing abilities. This assumption is far from the reality and considering the effects of false alarm P_{fa} and miss-detection P_{md} may greatly undermine our findings. Consequently, the performance of proposed algorithms in Section III-A should be evaluated under more realistic conditions. In [15], it has been shown that the learning algorithm that is used to update the probability distribution in local nodes is completely robust to P_{fa} and P_{md}. To guarantee that the performance of the sensing algorithms will not be significantly degraded by considering the effects of false alarm and miss-detection, we increase the threshold of selecting an edge. Increasing the threshold makes the underlying graph more dense and increases the chance that correctly sensed channels are chosen for future sensing attempts. For sake of simplicity, we assume that all SUs have the same probability of false alarm as well as the same probability of miss-detection. Therefore, the threshold of probability for connecting an edge between any two vertices should be increased. The amount of increase in $p(\delta)$ to compensate the effect of miss-detection, $\overline{p}(\delta)$, is

$$\overline{p}(\delta) = (1 - p(\delta))P_{md}. \tag{11}$$

The effect of false alarms can be seen as an increase in missing opportunities and reduction in the algorithm's performance.

IV. Experimental Results

In this section, we provide the simulation results for findings in Sections III. The simulation parameters for this experiment are given in Table I. In Table I, N_{PU_0} represents the number of PU channels that initially are busy.

TABLE I
SPECTRUM SENSING SIMULATION PARAMETERS

Parameter	Value	Parameter	Value	Parameter	Value
n	$[1, \ldots, 15]$	a_{00}	0.9	N_{PU_0}	5
m	10	a_{11}	0.9	T	1000

In Figure 4 the performances of Algorithms 1 and 2 is compared with the non-cooperative and the parallel cooperative scheme [9]. We have depicted the sampling success rate versus the number of SUs. Sampling success rate is the ratio of the number of successful sensing attempts to the total number of sensing attempts in all SUs. In non-cooperative (greedy) setup, no cooperation is carried out among the sensing nodes. Each node based on its own history, which is generated locally using *Alg1* [15], decides to sense the best possible channel. The drawback of this non-cooperative method is missing the opportunities. Consider the scenario that a few PU channels are observing an empty channel for a long period of time, this causes the adaptive algorithm to repeatedly increase the probability of choosing that channel for sensing for all of the SUs. Obviously, SUs have to compete for those good channel and the unsuccessful SUs will remain with no available channel although, it may also exist some relatively worse PU channels that are empty during that time slot. These channels are sensed by none of the SUs due to greedy sensing algorithm which causes inefficient sensing diversity.

In parallel cooperative sensing a set of SUs are assigned to sense the spectrum and find an empty PU channel. To have a fair comparison, we calculated the success rate as the ratio of success of all of selected SU in one attempt to total attempts for this case. The number of sensing SUs are optimized according to the rate maximization analysis that is provided in [9].

In Figure 4, when the number of SUs is small the performance gap between the cooperative and the non-cooperative algorithms are considerable while increasing the number of SUs and consequently decreasing the opportunities this gap starts closing. As it can be seen in Figure 4, the parallel cooperative scheme [9] performs better than Algorithm 1 which is based on thresholding. It should be noted that in Algorithm 1 only one sensing attempt in one specific SU is performed. While in parallel sensing [9] for finding an empty channel, multiple SUs perform the sensing. It can be seen that by increasing the number of SUs, the chance of finding opportunities decreases in parallel sensing scheme and falls below Algorithm 1. In addition the communication overhead among the SUs and CN in Algorithm 1 is minimized that can be very important factor in large CR networks.

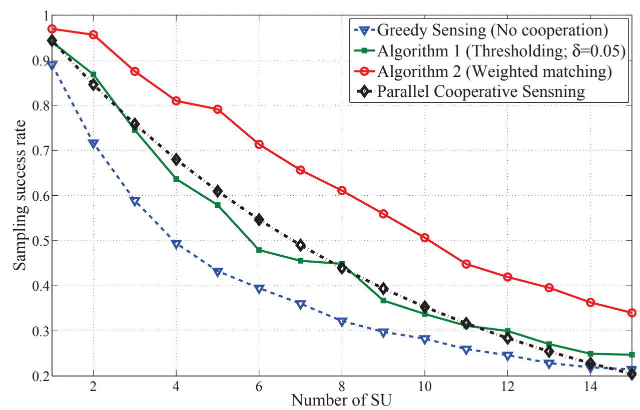

Fig. 4. Performance of Sensing algorithms with different matching scheme versus the greedy sensing method.

In Figure 5, we have obtained the values of optimal threshold over the probability of connecting an edge for Algorithm 1. We have used (2) to find the values of threshold by defining two confidence levels, $p_f = 95\%$ and $p_f = 99\%$, for the graphs to satisfy the Hall's theorem condition. We have shown the optimal threshold versus the different number of PU channels, m. For this experiment, we have assumed PU activity statistics are similar to Table I and for each point in the graph the value of n is assume equal to the average number of free PU channels (i.e. $n = m/2$).

For example, given $m = 20$, $n = 10$, $N_{PU_0} = 10$, and $A = \left[\begin{smallmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{smallmatrix}\right]$, we find the probability of having a graph that does not satisfy the Hall's theorem condition, p_f. This probability is given by (2). To find the exact value of p_f, we should define a threshold over the probability of connecting an edge. We set this threshold equal to $\delta = 0.2$. Therefore, using (10) and (2), we find $p_f = 0.13$ which means among all possible bipartite graphs 87% of the time the Hall's theorem condition holds and a complete matching can be found. The

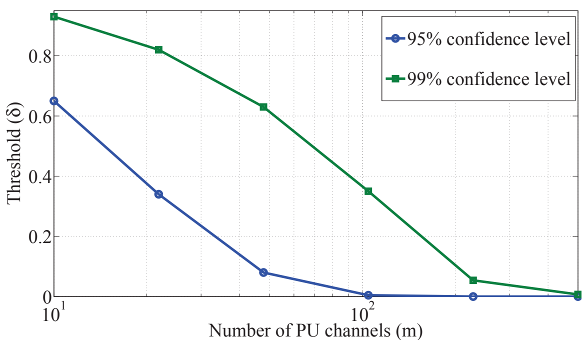

Fig. 5. Selecting the optimal threshold ($n = m/2$).

effects of non-ideal sensing are depicted in Figure 6. In this Figure, we have represented the performance of proposed algorithms considering the effects of miss-detection and false alarm probabilities. In this experiment the number of SUs are fixed (i.e. $n = 10$) and the rest of the parameters are equal to the values in Table I. For each algorithm two values of false-alarm probabilities are considered ($P_{fa} = [0.05, 0.1]$) and the sampling success rate are calculated versus different values of miss-detection probabilities. As we have expected, Algorithm 2 is degraded more than Algorithm 1 due to miss-detection. The reason for this observation is the ability of Algorithm 1 to increase the value of threshold based on (11), to avoid performance degradation. It can be inferred from Figure 6 that both of the proposed algorithms are robust to P_{md} and P_{fa}.

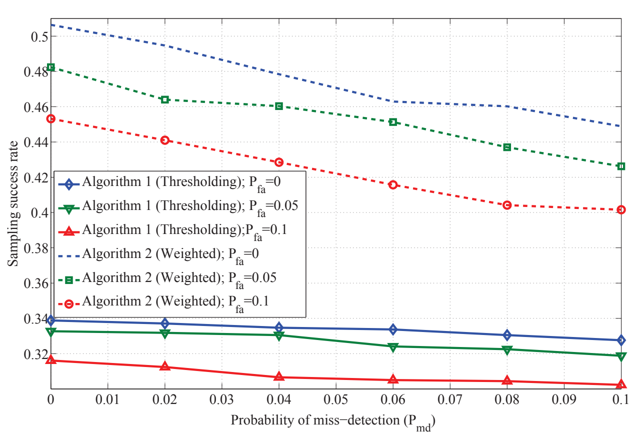

Fig. 6. The effects of false alarm and miss-detection on the sensing success rate of the proposed algorithms.

V. Concluding Remarks

In this paper, we have considered the problem of cooperative-parallel spectrum sensing in cognitive radio networks. The challenging conditions of implementing an agile radio requires efficient and effective spectrum sensing techniques that maximize the chance of finding empty channels and minimize the sensing time simultaneously. In this work we have combined parallel cooperation in cognitive radios with efficient sensing techniques to develop two cooperative-parallel spectrum sensing algorithms that can be used to jointly sense and allocate spectrum holes to the secondary users. These algorithms require a central node to perform the channel allocation for efficient sensing. The central node solves the corresponding bipartite matching problem and determines the optimal channel sensing for each secondary user. The results of this paper shows that employing the represented algorithms greatly increases the efficiency of channel sensing by increasing the success rate of sensing. It also increases the system efficiency by finding the maximal opportunities for sensing.

VI. Acknowledgement

This material is based upon work supported by the National Science Foundation under Grant No. CCF-0915994.

References

[1] 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.

[2] S. Haykin, "Cognitive radio: brain-empowered wireless communications," *Selected Areas in Communications, IEEE Journal on*, vol. 23, pp. 201–220, Feb. 2005.

[3] Y. Yuan, P. Bahl, R. Chandra, P. Chou, J. Ferrell, T. Moscibroda, S. Narlanka, and Y. Wu, "KNOWS: Cognitive radio networks over white spaces," in *New Frontiers in Dynamic Spectrum Access Networks (DYSPAN). 2nd IEEE International Symposium on*, pp. 416 –427, 2007.

[4] Q. Zhao, L. Tong, A. Swami, and Y. Chen, "Decentralized cognitive MAC for opportunistic spectrum access in ad hoc networks: A POMDP framework," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 3, pp. 589–600, 2007.

[5] T. Yucek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *Communications Surveys Tutorials, IEEE*, vol. 11, no. 1, pp. 116 –130, 2009.

[6] C. Cordeiro, K. Challapali, D. Birru, and N. Sai Shankar, "IEEE 802.22: the first worldwide wireless standard based on cognitive radios," in *New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005. 2005 First IEEE International Symposium on*, pp. 328 –337, 2005.

[7] D. Cabric, A. Tkachenko, and R. Brodersen, "Spectrum sensing measurements of pilot, energy, and collaborative detection," in *Military Communications Conference (MILCOM), 2006. IEEE*, pp. 1 –7, 2006.

[8] A. Min and K. Shin, "On Sensing-Access Tradeoff in Cognitive Radio Networks," in *New Frontiers in Dynamic Spectrum, IEEE Symposium on*, pp. 1–12, 2010.

[9] S. Xie, Y. Liu, Y. Zhang, and R. Yu, "A parallel cooperative spectrum sensing in cognitive radio networks," *Vehicular Technology, IEEE Transactions on*, vol. 59, no. 8, pp. 4079–4092, 2010.

[10] Z. Wang, Z. Feng, and P. Zhang, "An iterative hungarian algorithm based coordinated spectrum sensing strategy," *Communications Letters, IEEE*, vol. 15, pp. 49 –51, january 2011.

[11] Z. Bai, L. Wang, H. Zhang, and K. Kwak, "Cluster-based cooperative spectrum sensing for cognitive radio under bandwidth constraints," in *Communication Systems (ICCS), 2010 IEEE International Conference on*, pp. 569 –573, 2010.

[12] W.-Y. Lee and I. Akyildiz, "Optimal spectrum sensing framework for cognitive radio networks," *Wireless Communications, IEEE Transactions on*, vol. 7, no. 10, pp. 3845 –3857, 2008.

[13] G. Yuan, R. Grammenos, Y. Yang, and W. Wang, "Selective spectrum sensing and access based on traffic prediction," in *Personal, Indoor and Mobile Radio Communications, IEEE 20th International Symposium on*, pp. 1078 –1082, 2009.

[14] Y. Liu, R. Yu, and S. Xie, "Optimal cooperative sensing scheme under time-varying channel for cognitive radio networks," in *New Frontiers in Dynamic Spectrum Access Networks, 2008. DySPAN 2008. 3rd IEEE Symposium on*, pp. 1 –6, 2008.

[15] C. Tekin, S. Hong, and W. Stark, "Enhancing cognitive radio dynamic spectrum sensing through adaptive learning," in *Military Communications Conference, 2009. MILCOM 2009. IEEE*, pp. 1 –7, 2009.

[16] H. Kim and K. Shin, "Efficient discovery of spectrum opportunities with MAC-layer sensing in cognitive radio networks," *IEEE Transactions on Mobile Computing*, pp. 533–545, 2007.

[17] R. Diestel, "Graph theory, volume 173 of Graduate Texts in Mathematics," *Springer, Heidelberg*, vol. 91, p. 92, 2005.

[18] D. Burshtein and G. Miller, "Expander graph arguments for message-passing algorithms," *Information Theory, IEEE Transactions on*, vol. 47, pp. 782 –790, Feb. 2001.

[19] O. Damberg, S. Story, and T. Srevik, "A data parallel augmenting path algorithm for the dense linear many-to-one assignment problem," 1995.