

Maximizing System Throughput by Cooperative Sensing in Cognitive Radio Networks

Shuang Li*, Zizhan Zheng[†], Eylem Ekici[†] and Ness Shroff^{*†}

*Department of Computer Science and Engineering

Email: li.908@osu.edu

[†]Department of Electrical and Computer Engineering

The Ohio State University, Columbus, Ohio 43210

Email: {zhengz, ekici, shroff}@ece.osu.edu

Abstract—Cognitive Radio Networks allow unlicensed users to opportunistically access the licensed spectrum without causing disruptive interference to the primary users (PUs). One of the main challenges in CRNs is the ability to detect PU transmissions. Recent works have suggested the use of secondary user (SU) cooperation over individual sensing to improve sensing accuracy. In this paper, we consider a CRN consisting of a single PU and multiple SUs to study the problem of maximizing the total expected system throughput. We propose a Bayesian decision rule based algorithm to solve the problem optimally with a constant time complexity. To prioritize PU transmissions, we re-formulate the throughput maximization problem by adding a constraint on the PU throughput. The constrained optimization problem is shown to be strongly NP-hard and solved via a greedy algorithm with pseudo-polynomial time complexity that achieves strictly greater than 1/2 of the optimal solution. We also investigate the case for which a constraint is put on the sensing time overhead, which limits the number of SUs that can participate in cooperative sensing. We reveal that the system throughput is monotonic over the number of SUs chosen for sensing. We illustrate the efficacy of the performance of our algorithms via a numerical investigation.

I. INTRODUCTION

Cognitive radio networks (CRNs) have been proposed to address the spectrum scarcity problem by allowing unlicensed users (secondary users, SUs) to access licensed spectrum on the condition of not disrupting the communication of licensed users (primary users, PUs). To this end, SUs sense licensed channels to detect the primary user (PU) activities and find the underutilized “white spaces”. FCC has opened the TV bands for unlicensed access [3]¹, and IEEE has formed a working group (IEEE 802.22 [7]) to regulate the unlicensed access without interference. Many other organizations are also making efforts on the spectrum access policy in the CRN environment, e.g., DARPA’s ‘Next Generation’ (XG) program [17] mandates cognitive radios to sense signals and prevent interference to existing military and civilian radio systems. To avoid the interference to PUs, sensing becomes an indispensable part of CRN design.

Sensing can be performed via several methods, including energy detection, cyclostationary feature detection, and com-

pressed sensing [8]. Energy detection is a simple method and requires no a priori knowledge of PU signals [21]. Its main disadvantage is its decreased accuracy in face of fading, shadowing, and unknown noise power profiles. For instance, if an SU suffers from shadowing or heavy fading, the sensed signal tends to be weak while the PU is transmitting, leading to incorrect decisions. To address these problems while maintaining sensing simplicity, cooperative sensing schemes that fuse the sensing results of multiple SUs have been proposed [6][13][14].

Cooperative sensing overcomes shortcomings of individual sensing results by jointly processing observations. SUs in a locality report their individual sensing results, which are then used in a predefined decision rule to optimize an objective function. Examples of such functions include maximizing sensing accuracy (generally, a function of false alarm probability and mis-detection probability) or maximizing the system throughput. Aside from maximizing sensing accuracy related metrics, cooperative sensing schemes are also designed to estimate the maximum transmit power for SUs so that they do not cause disruptive interference to PUs [11]. On the other hand, cooperative sensing incurs additional sensing delay viz a viz individual sensing.

Three main categories of decision rules have been identified in [8]: *Soft Combining*, *Quantized Soft Combining*, and *Hard Combining*. In the first two categories, the sensing results are sent to the fusion center with little or no processing, while in the last one, binary local decisions are usually reported. Similar to sensor networks, linear fusion rules are widely applied to achieve a cooperative decision, such as AND, OR and majority rules [14]. In addition, a more advanced fusion technique that utilizes statistical knowledge [20] has been devised to capture the correlation between SUs in cooperative sensing. However, the resulting algorithm is suboptimal and its approximation factor is unknown. None of the above-mentioned works identify optimal decision rules for general decision structures and they require decision rules to assume particular forms (e.g., linear) for optimality analysis.

In this paper, we design an optimal data fusion rule to (hard) combining of the reported sensing result. More specifically, we aim to maximize the system throughput in a CRN composed of a single PU (i.e., single channel) and several SUs. While the

¹The recent FCC ruling requires the use of central TV Band usage databases to verify spectral availability. While respecting this ruling, our work explores local cooperative methods to improve sensing accuracy with the potential outcome of relieving this burdensome requirement.

target system is a simplified one, it is helpful in revealing the challenges associated with the design of optimal fusion rules. Moreover, the resulting algorithms can easily be generalized to more complex systems comprised of multiple channels, where sensing decisions are made per channel. Our main contributions can be summarized as follows:

- In contrast to previous works that restrict the class of fusion rules, we propose a Bayesian decision rule based algorithm to solve the throughput maximization problem optimally with constant time complexity.
- To guarantee resources for the PU, we re-formulate the problem by adding a constraint on the PU throughput. This constrained problem is shown to be strongly NP-hard by reducing the classical product partition problem [2] to it. A greedy algorithm is obtained with pseudo-polynomial time complexity. This approximation algorithm is analytically shown to achieve strictly greater than $1/2$ of the optimal solution.
- We investigate systems where limited sensing overhead is allowed, i.e., the number of sensing SUs is restricted. Our theoretical results show that the performance of cooperative sensing is monotonic over the number of SUs used for sensing.

The paper is organized as follows: Related work is presented in Section II. In Section III, the system model is introduced. The system throughput maximization problem is formulated in Section IV, and solved optimally via Bayesian decision rule. In Section V, the constrained maximization problem is formulated, which is shown to be strongly NP-hard. A pseudo-polynomial time greedy algorithm is proposed with an approximation factor strictly greater than $1/2$. Another direction is considered in Section VI where the system throughput is maximized subject to a constraint on the number of sensing SUs used. In Section VII, numerical results are presented for the performance of our algorithms. The paper is concluded in Section VIII.

II. RELATED WORK

Cooperative sensing solutions have been investigated in recent years. They rely on multiple SUs to exchange sensing results or a central controller to collect the sensing results from SUs. The network is usually divided into clusters and each cluster head makes the decision on the channel occupancy. Collaborations among SUs have been shown to improve the efficiency of spectrum access and allow the relaxation of constraints at individual SUs [4][24]. One branch of the papers in cooperative sensing assume that the length of sensing time at individual SUs is proportional to the sensing accuracy. However, longer sensing time decreases the transmission time. The trade-off is called the *sensing efficiency* problem and is discussed in [10] and [12]. In our work, we assume the observation time at each SU is fixed so that the individual sensing accuracy does not depend on it. We focus on the optimal decision rule based on the sensing results collected.

Decision rules so far mainly focus on AND, OR, majority rules and other linear rules. Zhang et. al. [25] show that the op-

timal fusion rule to minimize the cooperative sensing error rate is the half-voting rule in most cases. They show that AND or OR rules are optimal only in rare cases. However, other rules with more complicated forms have not been considered in [25]. Based on these observations, a fast spectrum sensing algorithm is proposed for a large network where not all SUs are required for sensing while satisfying a given error bound. However, the optimal number of sensing nodes and the complexity of this problem have not been discussed. In [14], the SU throughput is maximized subject to sufficient protection provided to PUs. The optimal k -out-of- N fusion rule is determined and the sensing/throughput trade-off is also analyzed. As in [25], no fusion rules of general forms are considered. Thus, the optimization is restricted to a small fusion rule domain. Shahid et. al. [18] consider the spatial variation of SUs and the fusion rule is a weighted combination of SU observations. The weight depends on the received power and path loss at each SU. Though more advanced than AND, OR, and majority rules, the weighted form is restricted to the linear function domain. In [5], optimal multi-channel cooperative sensing algorithms are considered to maximize the SU throughput subject to per channel detection probability constraints. The resulting non-convex problem is solved by an iterative algorithm. Compared to [5], our work focuses on the maximization of the system throughput, including the PUs and SUs. Although we only consider a single-channel network, which is a simplification made on the model, our decision algorithms can be applied for each channel individually. Moreover, a soft decision rule is considered in [5], which requires significant amount of data to be transmitted to the coordinator while our hard decision rule requires only one bit sent from each SU.

III. SYSTEM MODEL

We consider a time-slotted cognitive radio network in which a PU network, consisting of a PU base station (PU-BS) and PU receivers, co-exists in the same area with an SU base station (SU-BS) and M SUs (Figure 1). We focus on the PU transmissions over a particular channel. We consider uplink part for the SU system, i.e., only one SU can be active and transmit to the SU-BS at any given time. Some PU receivers may lie in the interference range of SUs such as PU 1 in Figure 1. Any transmission from these SUs such as SUs 1, 2, and 3 in Figure 1 may cause interference to those PU receivers. We denote the set of SUs whose uplink transmission causes interference to PU receivers by S and $|S| = N$ ($M \geq N$). They are indexed from 1 to N . SUs outside S can use the channel to transmit at any time slot without causing interference to the PUs.

SUs in S are close to the PU network and they may sense the channel cooperatively to reduce the sensing errors. The sensing results of individual SUs are assumed to be independent. Let B represent the PU activity such that $B = 1$ if PU is active, and $B = 0$ otherwise. Let P_f^i denote the **probability of a false alarm** for SU i , which is the probability that SU i senses the PU to be active given that the PU is actually idle. P_m^i represents the **probability of mis-detection** for SU i , which

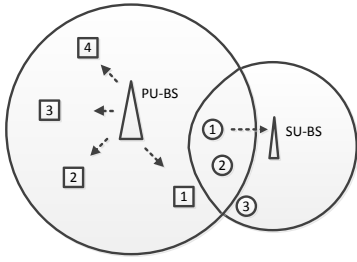


Fig. 1. System model of an SU network overlaid with a PU network.

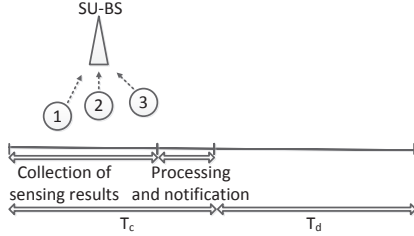


Fig. 2. Control slot T_c and data slot T_d .

is the probability that SU i senses the PU to be idle given that the PU is actually active.

Cooperative Sensing: Multiple SUs are chosen to sense the channel and the SU-BS predicts the PU activity by collecting the sensing results from these SUs. We denote the set of SUs that participate in the cooperative sensing as S_0 , where $|S_0| = k$. Note that $S_0 \subseteq S$. In the cooperative sensing model, we assume the SU-BS collects sensing results from SUs in S_0 .

Cooperative Sensing Indicator: The observation of the PU activity by SU i is denoted by o_i . $o_i = 1$ indicates that SU i observes the PU to be active, while $o_i = 0$ indicates that SU i observes the PU to be idle. In this paper, our objective is to characterize S_0 and estimate the PU activity based on observations from S_0 (called the decision rule). The decision rule is denoted as a function $f : \Omega^k \rightarrow \Omega$ where $\Omega = \{0, 1\}$. The observations form a vector \mathbf{o} , where $\mathbf{o} \in \Omega^k$ while the decision is denoted by O where $O \in \Omega$. The false alarm probability of cooperative sensing is denoted by $P_f^c = P(O = 1|B = 0)$. The mis-detection probability of cooperative sensing is denoted by $P_m^c = P(O = 0|B = 1)$. One time slot is divided into a control slot T_c and a data slot T_d where $T_c + T_d = 1$ (Figure 2). In the control slot, the SU-BS collects sensing results from S_0 and notifies an SU in S if the cooperative sensing result is “idle” ($O = 0$). If the PU is active (mis-detection), the PU transmission will be collided with the transmission from the SU. The length T_c of the control slot is regarded as the sensing overhead and assumed to be constant throughout the paper. It means that a fixed time period is allocated for cooperative sensing in each slot.

The uplinks of SUs in S are assumed to have the same capacity which is normalized to 1. If the decision of the cooperative sensing at the SU-BS is “idle”, the SU-BS notifies one of the SUs in S (not limited to S_0 , the sensing set) to transmit. We assume SUs in S are always backlogged. The scheduling of the transmitting SUs is beyond the scope of this paper. However, any work-conserving scheduling policy

TABLE I
NOTATION LIST

Symbol	Meaning
M	Total number of SUs in the secondary network
S	Set of SUs which cause interference to PU receivers
N	$ S $
S_0	Set of SUs that are chosen to sense the channel. $S_0 \subseteq S$
k	$ S_0 $
P_f^i	False alarm probability of SU i
P_m^i	Mis-detection probability of SU i
P_f^c	False alarm probability of cooperative sensing
P_m^c	Mis-detection probability of cooperative sensing
T_c	Control slot
T_d	Data slot
π_0	Probability that the PU is idle
γ	Average throughput of PUs in the interference range of a SU

operating on idle slots can be used together with the decision rule to maximize the total system throughput. We let π_0 denote the probability that the PU is idle and we assume that the prior distribution of PU activity is acquired over time accurately. Note that we do not restrict the PU activity to any specific distribution except that it does not change within one time slot. The average throughput of PUs whose transmission would be interfered by SUs in S is denoted as γ . Table I summarizes the notations used in the paper.

The SU communication follows a protocol with the following outline:

- 1) SUs report P_m^i 's and P_f^i 's to SU-BS;
- 2) SU-BS determines the sensing set S_0 and the decision rule f based on P_m^i , P_f^i 's and the optimization metric;
- 3) SU-BS notifies SUs in S_0 with an *ACK* and also assigns each one of them a *SEQ* number for reporting sensing results;
- 4) SUs receiving an *ACK* sense the channel and report the results to SU-BS in the order of *SEQ*;
- 5) SU-BS makes the decision of the PU activity based on the sensing results and f and schedules an SU for transmission if the decision is 0 (PU idle).

IV. SYSTEM THROUGHPUT MAXIMIZATION

In this section, we formulate the cooperative sensing problem with the assumption that $S_0 = S$, that is, the sensing results from all SUs in S are reported to SU-BS within T_c . SUs outside S can transmit without causing interference to the PUs. Thus, their performance is independent of the choice of the sensing set or the decision rule. Our goal is to maximize the sum of the expected throughput of SUs in S and that of the PUs whose transmission may be interfered by the SUs. It is equivalent to maximizing the expected throughput of the system with PU-SU co-existence.

A. Problem Formulation

Given $B = 0$ (the PU is idle), the probability of a particular observation vector \mathbf{o} occurring is

$$P(\mathbf{o}|B = 0) = \prod_{i \in S, o_i = 1} P_f^i \prod_{j \in S, o_j = 0} (1 - P_f^j). \quad (1)$$

The sum of all $P(\mathbf{o}|B=0)$'s with $f(\mathbf{o})=0$ is given as

$$P(O=0|B=0) = \sum_{f(\mathbf{o})=0} P(\mathbf{o}|B=0). \quad (2)$$

Then, the false alarm probability of cooperative sensing is

$$P_f^c = 1 - P(O=0|B=0) = 1 - \sum_{f(\mathbf{o})=0} P(\mathbf{o}|B=0). \quad (3)$$

Likewise, given $B=1$ (the PU is active), the probability of a particular observation vector \mathbf{o} occurring is

$$P(\mathbf{o}|B=1) = \prod_{i \in S, o_i=1} (1 - P_m^i) \prod_{j \in S, o_j=0} P_m^j. \quad (4)$$

The sum of all $P(\mathbf{o}|B=1)$'s with $f(\mathbf{o})=1$ is given as

$$P(O=1|B=1) = \sum_{f(\mathbf{o})=1} P(\mathbf{o}|B=1). \quad (5)$$

Then, the mis-detection probability of cooperative sensing is

$$P_m^c = 1 - P(O=1|B=1) = 1 - \sum_{f(\mathbf{o})=1} P(\mathbf{o}|B=1). \quad (6)$$

Note that Equation (2) is the conditional probability that SU-BS correctly identifies the PU activity when it is idle so that one SU could transmit successfully; Equation (5) is the conditional probability that SU-BS correctly detects the PU is active so that no SU would transmit and the PU could transmit successfully. Accordingly, the expected throughput of the SUs can be represented by

$$\begin{aligned} (1 - T_c)P(B=0, O=0) &= (1 - T_c)\pi_0 P(O=0|B=0) \\ &= (1 - T_c)\pi_0 \sum_{f(\mathbf{o})=0} P(\mathbf{o}|B=0), \end{aligned} \quad (7)$$

since the uplinks of SUs in S are assumed to have capacity 1 and only one of them could be scheduled in each time slot. The expected throughput of the PU can be represented by

$$\gamma P(O=1|B=1) = \gamma \sum_{f(\mathbf{o})=1} P(\mathbf{o}|B=1) \quad (8)$$

since γ is the average throughput of the PU whose transmission would be interfered by SUs in S . The problem is then formulated as follows:

Problem (A):

$$\max_f (1 - T_c)\pi_0 \sum_{f(\mathbf{o})=0} P(\mathbf{o}|B=0) + \gamma \sum_{f(\mathbf{o})=1} P(\mathbf{o}|B=1)$$

B. Optimal Solution with Bayesian Decision Rule

We show that Problem (A) can be converted to a Bayesian Decision problem. Algorithm 1 is then proposed based on Bayesian decision rule to minimize the posterior expected loss [1] and it is of constant time complexity.

Problem (A) is equivalent to Problem (9) in terms of optimal f .

$$\begin{aligned} \min_f L(B=0, O=1) &\left[\pi_0 \sum_{f(\mathbf{o})=1} P(\mathbf{o}|B=0) \right] \\ + L(B=1, O=0) &\left[(1 - \pi_0) \sum_{f(\mathbf{o})=0} P(\mathbf{o}|B=1) \right], \end{aligned} \quad (9)$$

where $L(B, O)$ is the loss of decision O based on observation

Algorithm 1 Bayesian Decision Rule Based Algorithm for maximizing the system throughput (given \mathbf{o} , decide O)

```

1: if  $(1 - T_c)\pi_0 \prod_{o_i=1} P_f^i \prod_{o_j=0} (1 - P_f^j) \geq$ 
    $\gamma \prod_{o_i=1} (1 - P_m^i) \prod_{o_j=0} P_m^j$  then
2:    $O \leftarrow 0$ 
3: else
4:    $O \leftarrow 1$ 

```

\mathbf{o} , which is a non-negative number. $L(B=0, O=1) = (1 - T_c)$ and $L(B=1, O=0) = \frac{\gamma}{1 - \pi_0}$. Thus Equation (9) is the *posterior expected loss* of decision O (Definition 8 of Chapter 4.4 in [1]). Using the Bayesian decision rule, Problem (9) can be solved optimally [1]: given \mathbf{o} , the decision $O=1$ if $L(B=0, O=1)\pi_0 P(\mathbf{o}|B=0) < L(B=1, O=0)(1 - \pi_0)P(\mathbf{o}|B=1)$ and $O=0$ otherwise. Algorithm 1 is designed accordingly.

V. GUARANTEEING A TARGET PU THROUGHPUT

In this section, we investigate the maximum throughput problem with a PU throughput constraint. With higher priority, a minimum PU throughput is guaranteed in the problem formulation. We first show that this constrained problem is strongly NP-hard by reducing the classical product partition problem [2] to it. Then a greedy approximation algorithm is proposed to achieve strictly greater than 1/2 of the optimal solution. The complexity of the algorithm is shown to be pseudo-polynomial by solving a two-dimensional dynamic programming problem.

A. Problem Formulation and Properties

We formulate the constrained optimization problem as follows:

Problem (B):

$$\begin{aligned} \max_f (1 - T_c)\pi_0 \sum_{f(\mathbf{o})=0} P(\mathbf{o}|B=0) + \gamma \sum_{f(\mathbf{o})=1} P(\mathbf{o}|B=1) \\ \text{s.t.} \quad \sum_{f(\mathbf{o})=1} P(\mathbf{o}|B=1) \geq \alpha. \end{aligned} \quad (10)$$

Equation (10) is the constraint we put on Problem (B) where the expected PU throughput must be no less than a preset system-dependent threshold. It is equivalent to $1 - \sum_{f(\mathbf{o})=1} \prod_{i \in S, o_i=1} (1 - P_m^i) \prod_{j \in S, o_j=0} P_m^j \leq 1 - \alpha$, where $1 - \alpha$ is the collision factor. This can be interpreted as the probability that a PU transmission colliding with an SU transmission being no greater than $1 - \alpha$. Problem (B) maximizes the expected system throughput given that the lowest PU throughput can be met considering the high priority of the PU in cognitive radio networks.

By observing the structure of Problem (B), we state Lemma 5.1 that shows the optimal assignment of observations with $G(\mathbf{o}) < H(\mathbf{o})$ where $G(\mathbf{o}) = (1 - T_c)\pi_0 P(\mathbf{o}|B=0)$ and $H(\mathbf{o}) = \gamma P(\mathbf{o}|B=1)$. We define f^* as the optimal solution to Problem (B).

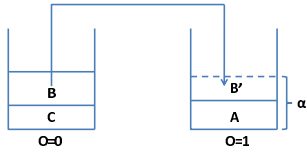


Fig. 3. Assignment of observations with no constraint and the optimal assignment for Problem (B). $A + B + C$ is the optimal assignment with no constraint while $A + B' + C$ is the optimal solution to Problem (B).

Lemma 5.1: In the optimal solution to Problem (B), we have $f^*(\mathbf{o}) = 1$ for all $G(\mathbf{o}) < H(\mathbf{o})$.

Proof: (Prove by contradiction) Assume that $f^*(\mathbf{o}) = 0$ for some \mathbf{o} where $G(\mathbf{o}) < H(\mathbf{o})$. Moving it from $O = 0$ to $O = 1$ increases $\sum_{f(\mathbf{o})=1} P(\mathbf{o}|B = 1)$ so that this operation still makes a feasible solution. Furthermore, the expected system throughput increases considering $G(\mathbf{o}) < H(\mathbf{o})$, which makes a better solution than the current optimal one. It causes a contradiction. Hence, we have $f^*(\mathbf{o}) = 1$ for all $G(\mathbf{o}) < H(\mathbf{o})$ in the optimal solution to Problem (B). ■

With the property of Lemma 5.1, we only need to decide which observations with $G(\mathbf{o}) \geq H(\mathbf{o})$ should be put in $O = 1$ to solve Problem (B) optimally. We define $\chi = \{\mathbf{o} : G(\mathbf{o}) \geq H(\mathbf{o}) \text{ and } f^*(\mathbf{o}) = 1\}$, which is the set of observations that need to be moved to $O = 1$ in the optimal solution; $\psi = \{\mathbf{o} : G(\mathbf{o}) \geq H(\mathbf{o}) \text{ and } f^*(\mathbf{o}) = 0\}$, which is the set of observations that stay in $O = 0$ in the optimal solution. Moreover, we define $A = \sum_{\mathbf{o}: G(\mathbf{o}) < H(\mathbf{o})} H(\mathbf{o})$, which is the contribution of observations with $G(\mathbf{o}) < H(\mathbf{o})$ in the optimal solution; $B = \sum_{\mathbf{o} \in \chi} G(\mathbf{o})$, which is the contribution of observations in χ when put in $O = 0$; $B' = \sum_{\mathbf{o} \in \chi} H(\mathbf{o})$, which is the contribution of observations in χ when put in $O = 1$ ($B \geq B'$); $C = \sum_{\mathbf{o} \in \psi} G(\mathbf{o})$, which is the contribution of observations in ψ when put in $O = 0$; $C' = \sum_{\mathbf{o} \in \psi} H(\mathbf{o})$, which is the contribution of observations in ψ when put in $O = 1$ ($C \geq C'$). Then, $A + B + C$ is the optimal solution to Problem (B) without the PU throughput constraint; $A + B' + C$ is the optimal solution to Problem (B), which is no greater than $A + B + C$. The optimal assignment is illustrated in Figure 3.

B. Proof of Strong NP-hardness

We focus on identifying the hardness of deciding observations with $G(\mathbf{o}) \geq H(\mathbf{o})$ that should be put in $O = 1$ in this section. It is shown to be strongly NP-hard by reducing the classical product partition problem [2] to the subproblem of it in Theorem 5.2.

Theorem 5.2: Problem (B) is strongly NP-hard.

Proof: We first state the classical product partition problem [2] - Given N positive integers: y_1, \dots, y_N , is there a way to have them partitioned into two equal-sized subsets that have the same product? For reduction, we construct an instance of Problem (B) by setting $(1 - T_c)\pi_0 = \gamma$, $\alpha = \epsilon + \sum_{\mathbf{o}: G(\mathbf{o}) < H(\mathbf{o})} H(\mathbf{o})$ with $\epsilon \leq \min_{\mathbf{o}: G(\mathbf{o}) \geq H(\mathbf{o})} H(\mathbf{o})$. For this instance, putting any \mathbf{o} with $G(\mathbf{o}) \geq H(\mathbf{o})$ to $O = 1$ would

make a feasible solution given that observations with $G(\mathbf{o}) < H(\mathbf{o})$ have all been put in $O = 1$. Obviously, choosing the observation with minimum non-negative $G(\mathbf{o}) - H(\mathbf{o})$ would be the optimal solution. Note that $G(\mathbf{o}) - H(\mathbf{o}) = 0$ is equivalent to $\frac{G(\mathbf{o})}{H(\mathbf{o})} = 1$. By setting $\frac{1 - P_f^i}{P_m^i} = \frac{1 - P_f^j}{P_m^j} = y_i$ for all i , we have $\frac{G(\mathbf{o})}{H(\mathbf{o})} = \prod_{\mathbf{o}_i=0, i=1, \dots, N} y_i \cdot \prod_{\mathbf{o}_j=1, j=1, \dots, N} \frac{1}{y_j}$. Now the instance becomes: given N pairs of integers $(y_1, \frac{1}{y_1}), \dots, (y_N, \frac{1}{y_N})$, exactly one number should be chosen from each pair; with this constraint, what is the minimum product no less than 1? The reduction from the product partition problem to this instance of Problem (B) can be done in polynomial time.

To verify the correctness of the reduction, we can check: if the minimum $\frac{G(\mathbf{o})}{H(\mathbf{o})}$ no less than 1 is 1, that is, the optimal solution of the instance is 1, we can answer ‘‘Yes’’ to the partition problem; if it is greater than 1, we can answer ‘‘No’’ to the partition problem. If Problem (B) can be solved in polynomial time, then the product partition problem can be solved in polynomial time as well. The product partition problem is well-known to be strongly NP-complete [2]. Assuming $P \neq NP$, Problem (B) has been proven to be strongly NP-hard. ■

It has been shown in Theorem 5.2 that finding the observation with $G(\mathbf{o})$ closest to $H(\mathbf{o})$ from above is strongly NP-hard. Hence, it is unlikely to find an efficient algorithm to solve Problem (B) optimally. We will focus on the approximation algorithm design in Section V-C.

C. Greedy Approximation Algorithm

We propose a greedy algorithm (Algorithm 2) that initially assigns all observations to $O = 1$ and then moves observations with $G(\mathbf{o}) \geq H(\mathbf{o})$ by $\frac{G(\mathbf{o})}{H(\mathbf{o})}$ from the highest to lowest to $O = 0$ until the feasibility constraint of Problem (B) is violated. By transforming Problem (B) into the Knapsack Problem [23], we will show that the algorithm achieves strictly greater than 1/2 of the optimal solution for Problem (B), which is $A + B' + C$ in Figure 3. Although the sum of $G(\mathbf{o})$ or $H(\mathbf{o})$ in the worst case has exponential number of terms, we will design a pseudo-polynomial time algorithm in Section V-D considering its combinatorial nature. Ignoring rounding errors, the implementation calculates these sums accurately.

In Algorithm 2, observations are chosen by $\frac{G(\mathbf{o})}{H(\mathbf{o})}$ from the highest to the lowest and assigned to $O = 0$ after those with $G(\mathbf{o}) < H(\mathbf{o})$ are assigned to $O = 1$. Ties are broken by putting observations with smaller $H(\mathbf{o})$ in the front. In Line 3, U is assigned to be the sum of contributions of all observations if put in $O = 1$ ($A + B' + C'$). In Lines 4-5, whether a feasible solution exists for the given input is checked by comparing the extreme case where all observations are assigned to $O = 1$ with the threshold $\alpha \times \gamma$. In Line 6, observations are initialized to $O = 1$. Lines 6-8 checks whether the feasibility constraint in Problem (B) has been satisfied under the initial assignment. If yes, observations with $G(\mathbf{o}) \geq H(\mathbf{o})$ are assigned to $O = 0$ by Bayesian decision rule. Lines 9-13 searches for observations with $G(\mathbf{o}) \geq H(\mathbf{o})$ from the highest $\frac{G(\mathbf{o})}{H(\mathbf{o})}$ to lowest until $sum2 + H(\mathbf{o}_i) \leq U - \alpha \times \gamma$ is violated (Line 12). Note

Algorithm 2 Greedy Approximation Algorithm for Problem (B)

Input: $N, T_c, \pi_0, \gamma, \alpha, P_m^i, P_f^i$ for all i

Output: f or “infeasible”

- 1: $G(\mathbf{o}) \leftarrow (1 - T_c)\pi_0 \prod_{i \in S, o_i=1} P_f^i \prod_{j \in S, o_j=0} (1 - P_f^j)$ for all \mathbf{o}
 - 2: $H(\mathbf{o}) \leftarrow \gamma \prod_{i \in S, o_i=1} (1 - P_m^i) \prod_{j \in S, o_j=0} P_m^j$ for all \mathbf{o}
 - 3: $U \leftarrow \sum_{\mathbf{o}} H(\mathbf{o})$
 - 4: **if** $U < \alpha \times \gamma$ **then**
 - 5: output “infeasible” and return
 - 6: $f(\mathbf{o}) \leftarrow 1$ for all \mathbf{o} , $sum1 \leftarrow \sum_{\mathbf{o}: G(\mathbf{o}) < H(\mathbf{o})} H(\mathbf{o})$
 - 7: **if** $sum1 \geq \alpha \times \gamma$ **then**
 - 8: $f(\mathbf{o}) = 0$ for all \mathbf{o} with $G(\mathbf{o}) \geq H(\mathbf{o})$ and return
 - 9: Sort \mathbf{o} 's with $G(\mathbf{o}) \geq H(\mathbf{o})$ in non-increasing order of $\frac{G(\mathbf{o})}{H(\mathbf{o})}$ and label them from 1 to l
 - 10: $sum2 \leftarrow 0$
 - 11: **for** $i = 1$ to l **do**
 - 12: **if** $sum2 + H(\mathbf{o}_i) > U - \alpha \times \gamma$ **then**
 - 13: **break**
 - 14: $sum2 \leftarrow sum2 + H(\mathbf{o}_i), f(\mathbf{o}_i) \leftarrow 0$
-

that $\sum_{\mathbf{o}: f(\mathbf{o})=0} H(\mathbf{o}) \leq U - \alpha \times \gamma$ and $\sum_{\mathbf{o}: f(\mathbf{o})=1} H(\mathbf{o}) \geq \alpha \times \gamma$ (feasibility constraint) are equivalent since $\sum_{\mathbf{o}} H(\mathbf{o}) = U$. $f(\mathbf{o})$ of these observations are set to be 0 (Line 13) in the searching process. Next, we state Theorem 5.3 that gives the approximation factor of Algorithm 2.

Theorem 5.3: Algorithm 2 achieves strictly greater than 1/2 of the optimal solution to Problem (B).

Proof: Recall the following notations: $A = \sum_{\mathbf{o}: G(\mathbf{o}) < H(\mathbf{o})} H(\mathbf{o})$, $B = \sum_{\mathbf{o} \in \mathcal{X}} G(\mathbf{o})$, $B' = \sum_{\mathbf{o} \in \mathcal{X}} H(\mathbf{o})$, $C = \sum_{\mathbf{o} \in \psi} G(\mathbf{o})$, $C' = \sum_{\mathbf{o} \in \psi} H(\mathbf{o})$, and $W = A + B + C$. We define $U = A + B' + C' = \sum_{\mathbf{o}} H(\mathbf{o})$. Let APX be the solution to Problem (B) output by Algorithm 2. Let OPT be the optimal solution to Problem (B). Then, we have

$$OPT = W - (B - B') \quad (11)$$

$$= A + C + B' = (A + B' + C') + (C - C') = U + (C - C').$$

Equation (11) holds by Figure 3 since moving observations from $O = 0$ to $O = 1$ loses $B - B'$ in throughput compared to the optimal solution with no constraint, which is W . Note that $(C - C')$ is the optimal solution to Problem (12) and it is the difference of contribution to throughput between keeping observations with $G(\mathbf{o}) \geq H(\mathbf{o})$ and $f^*(\mathbf{o}) = 0$ in $O = 0$ and moving them to $O = 1$.

$$\begin{aligned} & \max \sum_{G(\mathbf{o}_i) \geq H(\mathbf{o}_i), i=1, \dots, l} (G(\mathbf{o}_i) - H(\mathbf{o}_i)) x_i \\ \text{s.t.} \quad & \sum_{G(\mathbf{o}_i) \geq H(\mathbf{o}_i), i=1, \dots, l} H(\mathbf{o}_i) x_i \leq U - \alpha \times \gamma \quad (12) \\ & x_i \in \{0, 1\} \text{ for all } i, \end{aligned}$$

Algorithm 3 Pseudo-Polynomial Algorithm to Find the Joint Distribution of $(\log \frac{G(\mathbf{o})}{H(\mathbf{o})}, \log H(\mathbf{o}))$

Input: N, P_f^i, P_m^i for all i

Output: $C(N, j, j')$ for all j, j'

- 1: $y_i \leftarrow \text{round}(\log \frac{1 - P_f^i}{P_m^i}, r) \times 10^r$ for all i
 - 2: $z_i \leftarrow \text{round}(\log \frac{P_f^i}{1 - P_m^i}, r) \times 10^r$ for all i
 - 3: $\lambda_i \leftarrow \text{round}(\log P_m^i, r) \times 10^r$ for all i
 - 4: $\mu_i \leftarrow \text{round}(\log (1 - P_m^i), r) \times 10^r$ for all i
 - 5: $M \leftarrow \sum_{i=1}^N \max \{y_i, z_i\}, m \leftarrow \sum_{i=1}^N \min \{y_i, z_i\}$
 - 6: $M' \leftarrow \max \{\max_i \lambda_i, \max_i \mu_i\}, m' \leftarrow \sum_{i=1}^N \min \{\lambda_i, \mu_i\}$
 - 7: $C(i, j, j') \leftarrow 0$ for all i, j, j' , $C(1, y_1, \lambda_1) \leftarrow 1$, $C(1, z_1, \mu_1) \leftarrow 1$
 - 8: **for** $i = 1$ to $N - 1$ **do**
 - 9: **for** $j = m$ to M **do**
 - 10: **for** $j' = m'$ to M' **do**
 - 11: $C(i+1, j, j') = C(i, j - y_{i+1}, j' - \lambda_{i+1}) + C(i, j - z_{i+1}, j' - \mu_{i+1})$
-

where observations with $G(\mathbf{o}) \geq H(\mathbf{o})$ are labeled in an arbitrary order. Next, we will show that the constraint of Problem (12) and that of Problem (B) are equivalent. As shown in Figure 3, we have

$$A + B' \geq \alpha \times \gamma \Leftrightarrow A + B' - \alpha \times \gamma \geq 0$$

$$\Leftrightarrow A + B' + C' - \alpha \times \gamma \geq C' \Leftrightarrow C' \leq U - \alpha \times \gamma. \quad (13)$$

Problem (12) is a Knapsack Problem [23]. By the following greedy approach, at least 1/2 of $(C - C')$ can be achieved [23]: choosing observations with $G(\mathbf{o}_i) \geq H(\mathbf{o}_i)$ from the highest $\frac{G(\mathbf{o}_i) - H(\mathbf{o}_i)}{H(\mathbf{o}_i)}$ to the lowest until (12) is violated, which is exactly what we do in Algorithm 2 since $\frac{G(\mathbf{o}_i) - H(\mathbf{o}_i)}{H(\mathbf{o}_i)} \geq \frac{G(\mathbf{o}_j) - H(\mathbf{o}_j)}{H(\mathbf{o}_j)}$ if and only if $\frac{G(\mathbf{o}_i)}{H(\mathbf{o}_i)} \geq \frac{G(\mathbf{o}_j)}{H(\mathbf{o}_j)}$. Hence, $APX = U + 1/2(C - C')$ holds. Since $U > 0$, we always have $APX/OPT > 1/2$ for Problem (B). ■

So far, we have shown that the greedy approximation algorithm (Algorithm 2) exists for Problem (B) with an approximation factor strictly greater than 1/2. When $U \gg C - C'$, this factor could be arbitrarily close to 1.

D. Pseudo-Polynomial Implementation

In Lines 3, 6, 9 and 12 of Algorithm 2, exponential number of observations are involved in the worst case due to its combinatorial nature. We design a pseudo-polynomial time algorithm by means of dynamic programming for the implementation. The running time of a pseudo-polynomial time algorithm is polynomial in the numeric value of the input, which is exponential in the length of them assuming they are rational numbers [23]. For simplicity, we assume $(1 - T_c)\pi_0 = \gamma$ in Algorithm 3, which can though be extended to general cases without the assumption easily.

In Algorithm 3, dynamic programming is applied to calculate the joint distribution of $\log \frac{G(\mathbf{o})}{H(\mathbf{o})}$ and $\log H(\mathbf{o})$, which

counts the number of observations with the same $\log \frac{G(\mathbf{o})}{H(\mathbf{o})}$ and the same $\log H(\mathbf{o})$. Note that this algorithm does not require future information - only the collection of all sensing results from SUs in the current time slot is required. Lines 3, 6, 9 and 12 of Algorithm 2 can be calculated based on these counts. $\text{round}(a, r)$ rounds a to r decimal places. We use $\text{round}(a, r) \times 10^r$ to scale and round a real a to an integer. M and m specify the maximum and minimum contribution an observation \mathbf{o} can have to $\log \frac{G(\mathbf{o})}{H(\mathbf{o})}$ respectively, while M' and m' specify the maximum and minimum contribution an observation \mathbf{o} can have to $\log H(\mathbf{o})$ respectively. $C(N, j, j')$ records the number of observations with $\log \frac{G(\mathbf{o})}{H(\mathbf{o})}$ (after rounding) equal to j and $\log H(\mathbf{o})$ (after rounding) equal to j' . Boundary conditions are set in Line 7. Lines 8-10 use iterations to find $C(i, j, j')$ for all $i = 1, \dots, N$, $m \leq j \leq M$ and $m' \leq j' \leq M'$. The recursive function in Line 10 matches the fact that when the 0 observation from SU $i+1$ is added to observations from SU 1 to i , $\log \frac{G(\mathbf{o})}{H(\mathbf{o})}$ is added by y_{i+1} and $\log H(\mathbf{o})$ is added by λ_{i+1} ; on the other hand, when the 1 observation from SU $i+1$ is added to observations from SU 1 to i , $\log \frac{G(\mathbf{o})}{H(\mathbf{o})}$ is added by z_{i+1} and $\log H(\mathbf{o})$ is added by μ_{i+1} . Note that Line 10 may encounter $C(i, j, j')$ beyond the boundaries of j or j' , the value of which will be treated as 0. The time complexity is $O(N(M-m)(M'-m'))$, which is pseudo-polynomial.

After the $C(N, j, j')$ distribution is found, Lines 3, 6, 9 and 12 of Algorithm 2 can be calculated accordingly, the time complexity of which is dominated by $O(N(M-m)(M'-m'))$.

VI. SENSING SET IDENTIFICATION

In this section, we formulate a new problem where the SU-BS is free to choose any subset of S as the sensing set and maximizes the expected throughput of the system. We define d as the homogeneous reporting delay of the sensing results from an SU to the SU-BS, and \tilde{d} as the miscellaneous delay which covers all the processing required after the collection of sensing results at the SU-BS in T_c . No matter how many SUs are chosen in the sensing set, we always allocate the length of T_c as the control slot. Thus, the control overhead is still a constant in this section. Two cases are considered: $Nd + \tilde{d} \leq T_c$, where all SUs are allowed in the sensing set (Section VI-A); $Nd + \tilde{d} \geq T_c$, where at most $k = \lfloor \frac{T_c - \tilde{d}}{d} \rfloor$ SUs are allowed in the sensing set (Section VI-B). We show that the system throughput is monotonic over the number of SUs chosen in S_0 in Proposition 6.1. The hardness of the constrained problem with $Nd + \tilde{d} \geq T_c$ is unknown and there is no efficient algorithm proposed for this type of problem so far.

A. $Nd + \tilde{d} \leq T_c$

Without the constraint of the number of SUs in S_0 , we will show that the full set gives the most information. First, we define

$$P(\mathbf{o}^{S_0} | B = 0) = \prod_{i \in S_0, o_i=1} P_f^i \prod_{j \in S_0, o_j=0} (1 - P_f^j), \quad (14)$$

which is the probability of a particular observation vector \mathbf{o}^{S_0} where S_0 is the sensing set occurring given $B = 0$, and

$$P(\mathbf{o}^{S_0} | B = 1) = \prod_{i \in S_0, o_i=1} (1 - P_m^i) \prod_{j \in S_0, o_j=0} P_m^j, \quad (15)$$

which is the probability of a particular observation vector \mathbf{o}^{S_0} where S_0 is the sensing set occurring given $B = 1$. Then we formulate the problem as follows:

Problem (C):

$$\begin{aligned} \max_{f, S_0} (1 - T_c) \pi_0 \sum_{f(\mathbf{o})=0} P(\mathbf{o}^{S_0} | B = 0) \\ + \gamma \sum_{f(\mathbf{o})=1} P(\mathbf{o}^{S_0} | B = 1). \end{aligned}$$

Proposition 6.1: Let $F^*(S_0)$ be the optimal solution to Problem (C) with S_0 fixed. Then

$$F^*({i_1, \dots, i_k, i_{k+1}}) \geq F^*({i_1, \dots, i_k}). \quad (16)$$

Proof: (sketch) By adding the $k+1$ -th SU into the sensing set, we can at least achieve the same system throughput as before by ignoring its observation. The detailed proof can be found in our technical report [16]. ■

Using Proposition 6.1, we will prove it is the best choice to choose the full set as the sensing set in Corollary 6.2.

Corollary 6.2: For all $D \subset S$, we have $F^*(S) \geq F^*(D)$.

Proof: Given $D \subset S$, we index the elements in $S \setminus D$ from the smallest to the largest as l_1, \dots, l_m where $m = |S \setminus D|$ and $0 \leq m \leq N$. By Proposition 6.1, we have $F^*(D) \leq F^*(D \cup \{l_1\}) \leq F^*(D \cup \{l_1, l_2\}) \leq \dots \leq F^*(D \cup \{l_1, \dots, l_m\}) = F^*(S)$. ■

By Corollary 6.2, Problem (B) can be solved by first setting $S_0^* = S$ and then applying Algorithm 1 to find the optimal decision rule. Note that the time complexity is still $O(1)$.

B. $Nd + \tilde{d} > T_c$

We also investigate the case where the number of SUs in S_0 is constrained, and state that it is unlikely to have an efficient algorithm to find the optimal solution. By Proposition 6.1, the problem can be formulated as follows:

Problem (D):

$$\begin{aligned} \max_{f, S_0} (1 - T_c) \pi_0 \sum_{f(\mathbf{o})=0} P(\mathbf{o}^{S_0} | B = 0) \\ + \gamma \sum_{f(\mathbf{o})=1} P(\mathbf{o}^{S_0} | B = 1) \\ \text{s.t. } |S_0| = k, \end{aligned}$$

where $k = \lfloor \frac{T_c - \tilde{d}}{d} \rfloor$.

It has been shown in [15] that no non-exhaustive search method over the subset of S of size k can always solve it optimally when observations are correlated. For the independent observation problem such as Problem (D), however, it is not clear whether exhaustive search would be necessary as shown in [22]. Many heuristics such as Sequential Forward Selection (SFS, [19]), Sequential Backward Selection (SBS, [19]) and their variations [9] have been proposed to solve problems of this type. Although we characterize the monotonic property of system throughput over the number of SUs in the sensing set,

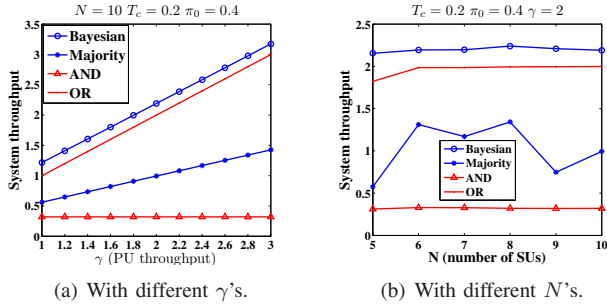


Fig. 4. Performance comparison of Bayesian decision rule, majority, AND and OR.

the complexity of the problem is not clear in the case when T_c is small, compared to N .

VII. SIMULATIONS

In this section, simulation results are presented for the performance of solutions proposed for Problems (A), (B), (C) and (D). We first compare the performance of Bayesian decision rule (Algorithm 1), majority, AND and OR policies [25] in Section VII-A. Then the performance of the greedy algorithm for Problem (B) (Algorithm 2), the random selection and the optimal solution are presented in Section VII-B. The performance of Sequential Forward Selection (SFS, [19]) is compared with the optimal solution to Problem (D) in Section VII-C. In all simulation studies, we consider a cognitive radio network with $N = 10$, $T_c = 0.2$, $\pi_0 = 0.4$, and $\gamma = 2$. For each parameter setting, we generate 30 groups of P_m^i 's and P_f^i 's randomly, which represents the random geographical locations of SUs in a CRN.

A. Performance of Bayesian Decision rule

Algorithm 1, the Bayesian decision rule based algorithm, has been proven to be optimal in Section IV. In Figure 4, we demonstrate the increase from majority, AND, OR rules in terms of system throughput, which is the objective function value of Problem (A). In majority rule, the decision is 1 only when the majority of the SUs sense an active PU; in AND rule, the decision is 1 only when all SUs sense an active PU; in OR rule, the decision is 1 if any of the SUs senses an active PU. We vary γ , the average PU throughput, and N , the number of SUs, respectively. Among all four algorithms, Bayesian decision rule strictly outperforms the other three. Among them, the OR rule is better than AND and majority rules since the PU transmission is better protected by the OR rule. OR rule is too conservative to guarantee SU transmission.

B. Performance of Greedy Algorithm

Greedy algorithm (Algorithm 2) can achieve strictly greater than $1/2$ of the optimal solution to Problem (B), as shown in Section V. We compare its performance with that of random selection. Random selection is also based on Bayesian decision rule, which means Algorithm 1 is first executed; after that, observations with $G(\mathbf{o}) \geq H(\mathbf{o})$ are randomly selected to put in $O = 1$ until the feasibility is satisfied. Thus the main difference between greedy algorithm and random selection lies

in the selection criterion of observations with $G(\mathbf{o}) \geq H(\mathbf{o})$ after the initial assignment based on Bayesian decision rule.

In addition, we set $\alpha = 0.8$, and $r = 2$. We vary parameters such as γ , the average PU throughput, α , the PU throughput constraint, and N , the number of SUs in Figure 5. To show the approximation factor of our algorithm accurately, two boundary cases are excluded in the result presentation where both greedy algorithm and random selection will give the optimal solution: 1) Bayesian decision rule gives the optimal solution; 2) It is optimal to put all observations in $O = 1$. Hence, we only show their performance when at least one but not all observations with $G(\mathbf{o}) \geq H(\mathbf{o})$ have to be moved to $O = 1$. The average case and worst case performances are calculated based on the results after the exclusion.

In Figure 5(a), the approximation factors of greedy algorithm and random selection over the optimal solution are compared over different values of γ , the average PU throughput in the system. With a higher γ , the factor decreases gradually in both average and worst cases of Algorithm 2 and it is the same with random selection although it fluctuates a bit due to the random selection. Greedy algorithm outperforms random selection in both average and worst cases. Potentially, the Bayesian decision rule assigns more SUs to $O = 1$ compared to a lower γ case. Thus the initial assignment is closer to α , the PU throughput constraint. Since we only consider cases where Bayesian decision rule is not optimal, both algorithms tend to have worse performance when the initial assignment approaches α because it gets more sensitive to a wrong observation selection.

In Figure 5(b), we vary α , the PU throughput constraint, and compare the performance of greedy algorithm and random selection. Greedy algorithm is obviously better than random selection in both average and worst cases. Furthermore, the worst performance of all random runs generated in greedy algorithm wins over the average performance in random selection. Approximation factors in both of them increase, although it is minor in greedy algorithm. The increase can be explained similarly to that in Figure 5(a): a higher α makes the initial Bayesian decision assignment farther away from it so that the performance is less sensitive to the choice of observations. Due to its randomness, random selection may have poor performance with the factor as low as about 0.4 in our simulation.

We test the performance of greedy algorithm with different scales of the network. The number of SUs is varied from 5 to 10. The approximation factors of both algorithms degrade with more SUs. However, the factor is always far above $1/2$, as proved in Theorem 5.3. The random selection drops below $1/2$ in some cases as shown in the figure.

C. Performance of Sequential Forward Selection

With $Nd + \tilde{d} \leq T_c$, Problem (C) is the same as Problem (A) in that the optimal sensing set is the full set (Corollary 6.2). Thus, we focus on the performance of SFS, a heuristic for Problem (D) whose hardness is unknown so far. In SFS, we start from an empty sensing set. In every step, only the SU

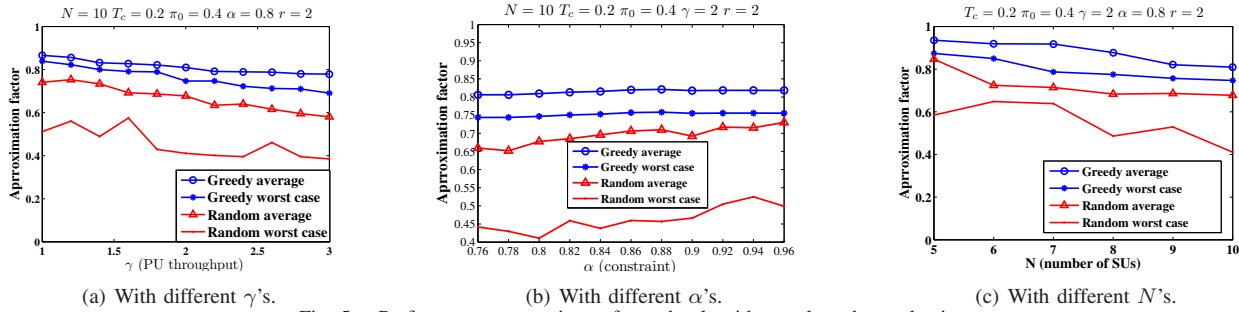


Fig. 5. Performance comparison of greedy algorithm and random selection.

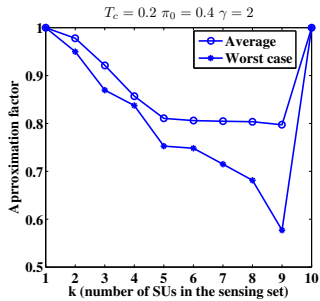


Fig. 6. Performance comparison of SFS over different k 's.

that is not yet chosen and has the largest marginal increase to the system throughput is added to the set. The algorithm stops when the size of the set reaches k . In Figure 6, we vary k , the size of the sensing set, from 1 to N and show the approximation factor of SFS compared to the optimal solution to Problem (D). When k increases, the performance of SFS degrades until $k = N$ where all SUs are chosen in the sensing set so that the order of selection does not matter. SFS on average achieves at least 0.8 of the optimal solution in our simulation although the factor is lower than 0.6 in one of the worst cases.

VIII. CONCLUSION

In this paper, we propose a series of algorithms to maximize the system throughput by cooperative sensing in cognitive radio networks. Bayesian decision rule is applied to solve the unconstrained optimization problem optimally. With the PU throughput constraint, the new problem is shown to be strongly NP-hard and a greedy approximation algorithm with pseudo-polynomial time complexity is proposed. More importantly, the approximation factor is strictly greater than 1/2. By restricting the number of SUs chosen for sensing, a new constrained optimization problem is formulated. We present a structural property that more SUs lead to better performance. However, the characterization of the combinatorial problem remains elusive, which is our future work. Moreover, we are also interested in investigating cases where the observations of SUs are correlated.

REFERENCES

[1] J. O. Berger. *Statistical Decision Theory and Bayesian Analysis*. Springer-Verlag, New York, 1985.
 [2] C. T. Ng, M.S.Barketau, T.C.E. Cheng and M. Y. Kovalyov. "Product Partition" and related problems of scheduling and systems reliability: Computational complexity and approximation. *European Journal of Operational Research*, 207(2):601–604, dec 2010.

[3] Federal Communications Commission. Notice of proposed rulemaking, in the matter of unlicensed operation in the tv broadcast bands (docket no. 04-186) and additional spectrum for unlicensed devices below 900 mhz and in the 3 ghz band (02-380), fcc 04-113. May 2004.
 [4] T. Do and B.L. Mark. Joint spatial-temporal spectrum sensing for cognitive radio networks. In *In proc. of CISS*, pages 124–129, march 2009.
 [5] R. Fan and H. Jiang. Optimal multi-channel cooperative sensing in cognitive radio networks. *IEEE Transactions on Wireless Communications*, 9(3):1128–1138, 2010.
 [6] Y.G. Li G. Ganesan. Cooperative spectrum sensing in cognitive radio networks. In *In proc. of DySPAN*, pages 137–143, Nov. 2005.
 [7] <http://grouper.ieee.org/groups/802/22/>. Ieee 802.22, working group on wireless regional area networks (wran).
 [8] R. Balakrishnan I. F. Akyildiz, B. F. Lo. Cooperative spectrum sensing in cognitive radio networks: A survey. *Physical Communication*, March 2011.
 [9] M. Kudo. Comparison of algorithms that select features for pattern classifiers. *Pattern Recognition*, 33(1):25–41, jan 2000.
 [10] W. Lee. Optimal spectrum sensing framework for cognitive radio networks. *IEEE Transactions on Wireless Communications*, 7(10):3845–3857, 2008.
 [11] B. Mark and A. Nasif. Estimation of maximum interference-free power level for opportunistic spectrum access. *Wireless Communications, IEEE Transactions on*, 8(5):2505–2513, may 2009.
 [12] A. Min and K. Shin. An optimal sensing framework based on spatial rssi-profile in cognitive radio networks. In *In proc. of SECON*, pages 207–215, Piscataway, NJ, USA, 2009. IEEE Press.
 [13] S. Mishra, A.t Sahai, and R. W. Brodersen. Cooperative sensing among cognitive radios. In *In proc. of ICC*, pages 1658–1663, 2006.
 [14] E.C.Y. Peh, Y. Liang, Y. Guan, and Y. Zeng. Optimization of cooperative sensing in cognitive radio networks: A sensing-throughput tradeoff view. *Vehicular Technology, IEEE Transactions on*, 58(9):5294–5299, nov. 2009.
 [15] J. M. Pena and R. Nilsson. On the complexity of discrete feature selection for optimal classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32:1517–1522, 2010.
 [16] S. Li, Z. Zheng, E. Ekici, and N. Shroff. Technical report. Maximizing System Throughput by Cooperative Sensing in Cognitive Radio Networks, 2011. http://www.cse.ohio-state.edu/~lish/Shuang_Li_TR_2011.pdf
 [17] F.W. Seelig. A description of the august 2006 xg demonstrations at fort a.p. hill. In *In proc. of DySPAN*, pages 1–12, april 2007.
 [18] M. Shahid and J. Kamruzzaman. Weighted soft decision for cooperative sensing in cognitive radio networks. In *In proc. of ICON*, page 6, 2010.
 [19] S. D. Stearns. On selecting features for pattern classifiers. In *In proc. of ICPR*, pages 71–75, Coronado, CA, 1976.
 [20] J. Unnikrishnan and V.V. Veeravalli. Cooperative sensing for primary detection in cognitive radio. *Selected Topics in Signal Processing, IEEE Journal of*, 2(1):18–27, feb. 2008.
 [21] H. Urkowitz. Energy detection of unknown deterministic signals. In *In proc. of the IEEE*, 55(4):523–531, april 1967.
 [22] J. M. Van Campenhout. *On the problem of measurement selection*. PhD thesis, Stanford University, 1978.
 [23] V. V. Vazirani. *Approximation Algorithms*. Springer, mar 2004.
 [24] E. Visotsky, S. Kuffner, and R. Peterson. On collaborative detection of tv transmissions in support of dynamic spectrum sharing. In *In proc. of DySPAN*, pages 338–345, nov. 2005.
 [25] Wei Zhang, R.K. Mallik, and K. Ben Letaief. Cooperative spectrum sensing optimization in cognitive radio networks. In *In proc. of ICC*, pages 3411–3415, may 2008.