Spectrum Sharing with Interference Coordination under Outage Probability Constraint

Jing Wang[†], Peili Cai[‡], Ping Zhang[†] and Guixia Kang[†]
[†]Key Laboratory of Universal Wireless Communication, Ministry of Education,
Beijing University of Posts and Telecommunications, Beijing, China
Email: wjcici@gmail.com
[‡]Department of Electrical and Computer Engineering, Texas A&M University
College Station, TX, USA

***Abstract*—In this paper we consider a cognitive radio (CR) communication system based on spectrum sharing schemes, where we have a secondary user (SU) link with multiple transmitting antennas and a single receiving antenna, coexisting with a primary user (PU) link with a single receiving antenna. At the SU transmitter (SU-Tx), the channel state information (CSI) of the SU link is assumed to be perfectly known; while the interference channel from the SU-Tx to the PU receiver (PU-Rx) is not perfectly known due to less cooperation between the SU and the PU. Considering a SU transmit power constraint, our design objective is to determine the transmit covariance matrix that maximizes the SU rate, while we protect the PU by enforcing both a PU average interference constraint and a PU outage probability constraint. This problem is formulated as a non-convex optimization problem with a non-explicit probabilistic constraint, which is then approximated as a mixed binary integer programming (MBIP) problem and solved with the Branch and Bound (BB) algorithm. The complexity of the BB algorithm is analyzed and numerical results are presented to show the performance of the CR system under consideration with our optimal solution.**

I. Introduction

The evolution from static spectrum allocation policies to dynamic ones can significantly increase the utilization efficiency of the radio spectrum. One promising platform to support such transitions is the cognitive radio (CR) system that was invented for spectrum sharing with existing primary links, where CRs dynamically adapt their transmission patterns to access under-utilized frequency segments while regulating the interference to PUs [1], [2]. As such, the key design challenge is how to maximize the SU rate while maintaining an acceptable level of interference to PUs.

Recently, there has been much research devoted to this interesting problem. The authors in [3] studied the channel capacity of a single secondary transmission when the interference power received at the PU-Rx is limited below a given threshold, which is the so-called interference temperature constraint. Along a similar line, the authors in [4] studied the problem of maximizing the sum utility over multiple SUs under the interference temperature constraints. In more recent research, the role of multi-antennas has been investigated under CR network settings. The authors in [5] studied the channel capacity of secondary multiple-input multiple-output (MIMO) and multiple-input single-output (MISO) channels when the CSI between the SU-Tx and PU is perfectly known at the SU-Tx. In the MISO case, under both an average secondary transmit power constraint and an interference temperature constraint at each PU-Rx, beamforming was proved to be the optimal strategy. In the MISO case where only one PU is present with one receiving antenna, a closed-form solution was derived. In [6] and [7], the authors considered a similar MISO scenario where, instead of complete CSI between the SU-Tx and PU, only partial CSI is known. In [6], channel capacity was studied with only the mean of the channel between the SU-Tx and PU-Rx is known at the SU-Tx, where beamforming was proved to be optimal. Such work was extended in [7] to consider both the mean and covariance feedbacks at the SU-Tx, where two algorithms are presented to solve for the optimal solution: one based on a second-order cone programming approach; and the other based on a geometric interpretation.

In this paper, we model a practical scenario where we know the distribution of the SU-Tx to PU-Rx channel and formulate the problem under an PU outage probability constraint in addition to the transmit power constraint and the average interference power constraint. In our work, we define the outage probability to be the probability of interference power at the PU-Rx exceeding a given threshold. The main motivation for this formulation is to allow some interference from the SU-Tx to the PU-Rx as long as the resulting outage probability is kept small. Our aim in this paper is to investigate the SU system performance with this more practical regulation over the SU interference to the PU-Rx. The main contribution of this paper is summarized as follows. We formulate the transmit covariance matrix design problem for a single secondary link under both an average interference constraint and an outage probability constraint to protect a given PU-Rx. Due to the introduction of the outage probability constraint, this resulting design problem is non-convex with non-explicit constraints. To solve this problem, we reformulate it into an MBIP problem with a deterministic constraint on the outage upper bound. Then we use a BB algorithm to compute the numerical results, which is highly efficient in solving the MBIP problem compared with exhaustive searching for the original non-convex problem.

The rest of the paper is organized as follows. In Section II,

978-1-936968-19-0 © 2011 ICST
DOI 10.4108/icst.crowncom.2011.245886

we discuss the system and signal models. In Section III, the MBIP transformation is discussed along with the BB algorithm and the complexity analysis. In Section IV, the numerical results are presented. Section V draws the conclusions.

Notations: $\mathbf{x}^\dagger$ denotes the conjugate transpose, $tr(\cdot)$ denotes the trace operator, $rank(\cdot)$ denotes the rank of a matrix, $E[\cdot]$ denotes the statistical expectation, and $C^{M\times N}$ denotes the space of $M\times N$ matrices with complex entries. Boldface upper and lower case letters are used to denote matrices and vectors, respectively, with "$\sim$" standing for "distributed as". Re$(\cdot)$ and Im$(\cdot)$ represent the real and imaginary parts of the operand, respectively. The $\log(\cdot)$ functions are over base 2.

II. System and Signal Model

We consider a simple CR system, where one SU link and one PU link share the same spectrum simultaneously. Here the SU-Tx is equipped with M_t transmitting antennas, and both the secondary and primary receivers are each equipped with a single antenna, as illustrated in Fig. 1. We assume that the SU-Tx knows the MISO channel $\mathbf{h}\in C^{M_t\times 1}$ from the SU-Tx to the SU-Rx, which is randomly distributed according to $\mathbf{h}\sim\mathcal{CN}(0,\mathbf{I})$. The MISO interference channel from the SU-Tx to the PU-Tx, denoted as $\mathbf{g}\in C^{M_t\times 1}$, is not perfectly known to the SU-Tx due to less cooperation between the SU and the PU. Specifically, we assume that the SU-Tx knows that the interference channel $\mathbf{g}$ can take values from a finite set $\mathbf{G}=\{\mathbf{g_1},\mathbf{g_2},\cdots,\mathbf{g_N}\}$ with a corresponding probability set $\{p_1,p_2,\cdots,p_N\}$. Under these assumptions, the SU-Tx adapts the transmission rate, power, and spatial spectrum to maximize its own transmission rate, while maintaining the interference to the PU-Rx below a certain level. Such an interference regulation is achieved by enforcing a set of constraints over the SU transmit covariance matrix, which will be discussed later in details. The signal model for the system under consideration is given as

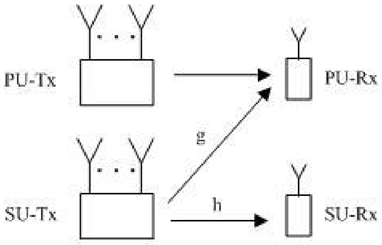

Fig. 1. System model of the SU coexisting with the PU

$$y=\mathbf{h}^\dagger\mathbf{x}+w, \tag{1}$$

where y and $\mathbf{x}\in C^{M_t\times 1}$ are the received and transmitted signals at the SU-Rx and SU-Tx, respectively, and w is the additive Gaussian noise with $w\sim\mathcal{CN}(0,1)$. The transmit covariance matrix is denoted by $\mathbf{K_x}=E[\mathbf{x}\mathbf{x}^\dagger]\succeq 0$.

Our goal in this paper is to balance the maximum transmit rate of the SU and the interference from the SU-Tx to the PU-Rx by adjusting the spatial spectrum of the SU signal. As such, we need to design the optimal transmit covariance matrix, $\mathbf{K_x}$, to maximize the SU rate with some tolerable interference to the PU-Rx. In particular, we cast this problem as follows:

$$\begin{aligned}
\text{(P1)}:\quad & \underset{\mathbf{K_x}}{\text{maximize:}} && \mathbf{h}^\dagger\mathbf{K_x}\mathbf{h} && (2)\\
& \text{subject to:} && tr(\mathbf{K_x})\leqslant P_{tr1} && (3)\\
& && E[\mathbf{g}^\dagger\mathbf{K_x}\mathbf{g}]\leqslant P_{tr2} && (4)\\
& && Pr\{\mathbf{g}^\dagger\mathbf{K_x}\mathbf{g}\geqslant r\}\leqslant p_{th} && (5)\\
& && \mathbf{K_x}\succeq 0. && (6)
\end{aligned}$$

where the objective is equivalent to maximizing the achievable rate $\log(1+\mathbf{h}^\dagger\mathbf{K_x}\mathbf{h})$, P_{tr1} is the SU transmit power limit, P_{tr2} is the average interference power limit, r is the instantaneous interference power tolerance at the PU-Rx, and p_{th} is the PU outage probability limit. The objective function is the SU transmission rate, and the four constraints are the average transmit power, the average interference power, the PU outage probability constraints, and the positive semi-definite constraint, respectively.

Due to the probabilistic constraint in (5), problem (P1) is generally hard to solve directly. For a probabilistic constraint where the random vector has a continuous distribution, checking the feasibility of each feasible point requires a complex multi-dimensional integration. Even when the random vector has a discrete distribution, the feasible set defined by the probabilistic constraint is generally non-convex and it cannot be described by explicit functions [8]. Fortunately, as shown in [9], [10], the above probabilistically constrained problems can be solved as integer programming (IP) problems with deterministic constraints.

For our problem, under the assumption that the SU-Tx knows that the interference channel $\mathbf{g}$ is of a finite discrete distribution, we take the approach in [9], [10] to first approximate (P1) as an MBIP problem with deterministic constraints, and then deploy a BB algorithm [11], [12], [13] to seek the solution. The details will be discussed in the next sections, together with complexity analysis and simulation results.

III. Optimization Algorithm

A. MBIP Transformation

In this section, we first discuss a deterministic transformation of the probabilistic constraint in (P1). As assumed, the random variable $\mathbf{g}$ takes values from a finite set $\mathbf{G}=\{\mathbf{g_1},\mathbf{g_2},\cdots,\mathbf{g_N}\}$ with a corresponding probability set $\{p_1,p_2,\cdots,p_N\}$. We refer to each probable value $\mathbf{g_n}$ as one scenario. The probabilistic constraint can then be interpreted as that the sum probability over all possible interference-violating scenarios must be bounded by p_{th}. Therefore, we can reformulate the probabilistic constraint in (P1) as shown in the following problem:

$$\text{(P2)}: \underset{\mathbf{K_x},\, b_n}{\text{maximize:}} \quad \mathbf{h}^{\dagger}\mathbf{K_x}\mathbf{h} \tag{7}$$

$$\text{subject to:} \quad tr(\mathbf{K_x}) \leqslant P_{tr1} \tag{8}$$

$$E[\mathbf{g}^{\dagger}\mathbf{K_x}\mathbf{g}] \leqslant P_{tr2} \tag{9}$$

$$\mathbf{g_n}^{\dagger}\mathbf{K_x}\mathbf{g_n} - Mb_n \leqslant r,\ n = 1, \cdots, N. \tag{10}$$

$$\sum_{n=1}^{N} b_n p_n \leqslant p_{th},\ b_n \in \{0, 1\} \tag{11}$$

$$\mathbf{K_x} \succeq 0. \tag{12}$$

The two newly added constraints (10) and (11) are deterministic and only involving explicit functions, which can be easily handled by numerical algorithms. The design variables here are now both the matrix $\mathbf{K_x}$ and the binary variables $b_n, n = 1, 2, \cdots, N$, where the binary variables are used to indicate whether the interference outage check needs to be performed: if $b_n = 0$, it means no outage is possible under the scenario $\mathbf{g_n}$ given the constraint (10), such that p_n needs not to be included in the left-hand sum of (11); if $b_n = 1$, there may or may not be an outage if the slack constant M is chosen large enough, which leads to the fact that (11) is enforcing an outage probability upper-bound to be less than p_{th} since p_n is now always counted in the left-hand sum of (11). The positive slack constant, M, is chosen to be of a large value since it is used to deactivate the outage check in (10) when $b_n = 1$. Given the fact that $\sum_{n=1}^{N} b_n p_n$ incurs an outage probability upper-bound, (P2) is actually a stricter version of (P1) with tighter constraints. As a result, the optimal objective value of (P2) will be slightly less than that of (P1). However, as we show later that the resulting performance is still much better than reference approaches.

we now discuss how to determine the value for M, which needs to guarantee the satisfaction of the inequality (10) when $b_n = 1$. For sufficiency, we could find an M that is larger than the maximum value of $\mathbf{g_n}^{\dagger}\mathbf{K_x}\mathbf{g_n}$ over $n = 1, ..., N$. One way to achieve that is as follows. Given that

$$\mathbf{g_n}^{\dagger}\mathbf{K_x}\mathbf{g_n} = tr(\mathbf{g_n}^{\dagger}\mathbf{K_x}\mathbf{g_n}) \tag{13}$$

$$= tr(\mathbf{K_x}\mathbf{g_n}\mathbf{g_n}^{\dagger}) \tag{14}$$

$$\leqslant tr(\mathbf{K_x})tr(\mathbf{g_n}\mathbf{g_n}^{\dagger}) \tag{15}$$

$$\leqslant P_{tr1}tr(\mathbf{g_n}\mathbf{g_n}^{\dagger}), \tag{16}$$

we take $M = \max_{n} P_{tr1}tr(\mathbf{g_n}\mathbf{g_n}^{\dagger})$.

With the value of M available, we next solve the MBIP problem (P2), for which a direct approach is through exhaustive search over the binary variables b_n's, where for each feasible choice of b_n's we solve the following convex semi-definite programming (SDP) problem

$$\underset{\mathbf{K_x}}{\text{maximize:}} \quad tr(\mathbf{K_x}\mathbf{h}\mathbf{h}^{\dagger}) \tag{17}$$

$$\text{subject to:} \quad tr(\mathbf{K_x}) \leqslant P_{tr1} \tag{18}$$

$$E[tr(\mathbf{K_x}\mathbf{g}\mathbf{g}^{\dagger})] \leqslant P_{tr2} \tag{19}$$

$$tr(\mathbf{K_x}\mathbf{g_n}\mathbf{g_n}^{\dagger}) \leqslant Mb_n + r,\ n = 1, \cdots, N. \tag{20}$$

$$\mathbf{K_x} \succeq 0. \tag{21}$$

Unfortunately, such an exhaustive search in general incurs exponential total complexity. So instead, we discuss a BB approach to search over the binary variables more efficiently in the average sense.

B. Branch and Bound Algorithm

As mentioned before, one way to solve an MBIP problem is through exhaustive search, where the feasible space grows exponentially with the number of binary variables, which leads to the NP-hardness of most binary optimization problems. Fortunately, BB algorithms [11], [12], [13] can often be used in solving discrete and combinatorial optimization problems to reduce the average complexity, when the problem has a finite but very large solution set with certain structures.

We first give a brief overview of the BB algorithm, followed by specific implementations for solving the MBIP problem (P2). Two components are usually required for an effective implementation of a BB algorithm. The first is a *branching procedure* and the second is a *bounding function*. Given a set S, the branching procedure returns non-overlapping subsets $S_1, S_2, ...$, whose union is the set S. The bounding function then computes the upper and/or lower bounds of the optimal value given each subset S_i. The upper and lower bounds are then used to determine one of the following two outcomes: split the subset S_i into more subsets for further bounding, or discard the subset S_i from the searching space, which is also referred to as pruning and is the reason why the BB algorithm is more efficient than exhaustive search.

It is clear that problem (P2) can be cast as a SDP problem over the design variable $\mathbf{K_x}$ when the binary variables are relaxed to be within $[0, 1]$. With this property, we next implement the BB algorithm to jointly search over $\mathbf{K_x}$ and the binary variable b_n. Due to the recursive nature of the BB algorithm, it traverses a binary search tree (BST), as shown in Fig. 2. Each *node* in the BST represents a particular case when the relaxed SDP problem from (P2) is solved at a partial or complete binary solution. In particular, the root node corresponds to the case where all b_n's are relaxed to be within $[0, 1]$; and a *leaf node* is a node at the bottom of the BST, which denotes the case with a complete binary solution, where the resulting objective value of (P2) is called an *incumbent* if it is the best objective value found so far across all the known leaf nodes. The *depth* of a node, D, is defined to be the number of determined binary variables in the partial binary solution at this node. As D increases from $D = j$ to $D = j+1$, one additional binary variable b_n is being determined. Specifically, at one particular node let us assume that $b_1, b_2, ..., b_{n-1}$ have been determined. We then create two child nodes corresponding to two sub-problems in the relaxed SDP form of (P2) with $b_n = 0$ and $b_n = 1$, respectively, while keeping $b_1, b_2, ..., b_{n-1}$ unchanged and rounding all undetermined binary variables, $b_{n+1}, b_{n+2}, ..., b_N$, to be ones. For a given sub-problem, if the achieved optimal objective value is lower than the current incumbent, the corresponding child node (as well as all of its descendants) is discarded, i.e., pruned from the searching space. Otherwise, the corresponding child node is kept in the BST, and the searching continues to b_{n+1} until we reach the leaf node with a complete binary solution. Following the above procedure, the BB algorithm traverses through the BST by solving one relaxed SDP for an optimal $\mathbf{K_x}$ at each node. The algorithm is terminated when the entire BST has been either pruned or processed. All computations in our algorithm are performed using the matlab-based software package CVX [14], [15] which deploys SeDuMi [16] as its back-end solver for SDP problems.

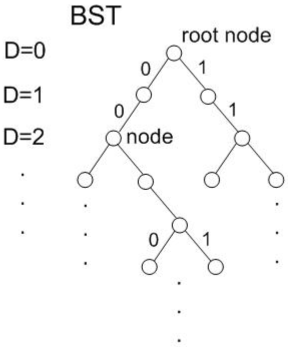

Fig. 2. Binary Search Tree (BST)

C. Complexity Analysis

In this section, we discuss the complexity of the proposed algorithm. The efficiency of the algorithm depends critically on the branching and bounding procedure, where the entire searching space is branched into non-overlapping subsets, and the bounding procedure then calculates bounds for each subset with decisions made on whether to continue branching or to discard the entire subset. The pruning process, which allows the algorithm to only traverse a fraction of the entire BST, is the key to decrease the overall searching complexity.

In our implementation, at the root node, there are no determined binary variables, i.e., all binary variables are relaxed. At each child node, one additional binary variable is determined. During each iteration, one node is chosen and the bound is calculated after solving the relaxed SDP. If the bound is lower than the incumbent, then it means that no child nodes branched from this node will yield a solution better than the incumbent; the node is therefore pruned. If the node at depth j is pruned, we can calculate how many potential child nodes of this branch are pruned, which indicates how much searching complexity is reduced. For simulations, we set $M_t = 2$. We assume that each element of $\mathbf{g_n}$ is generated by quantizing a random variable distributed as $\mathcal{CN} \sim (0, 0.1)$ into four levels, and the corresponding p_n is determined by integrating the probability density function over the associated quantization levels. The secondary transmit power ranges from 0 dB to 10dB. Accordingly, the MBIP problem has 16 binary design variables, such that if exhaustive search is deployed, there will be a total of $2^{16} = 65536$ sub-problems need to be solved. With our approach, Fig. 3 shows the update progress of the incumbent, and Fig. 4 shows the progress of pruned nodes in percentages at each iteration, where we only need to solve 273 sub-problems in this example.

$Remark$: The number of sub-problems solved in our BB algorithm varies over different channel realizations. Typically, we observe that less than 700 sub-problems in total are solved with our BB algorithm across a large number of channel realizations.

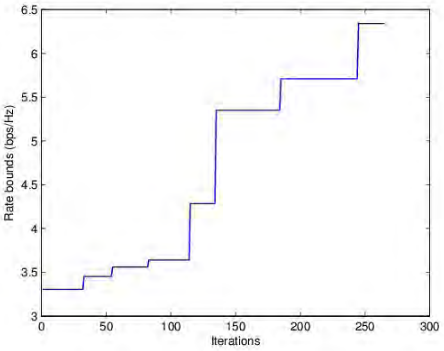

Fig. 3. Bounding progress of BB vs. the number of iterations

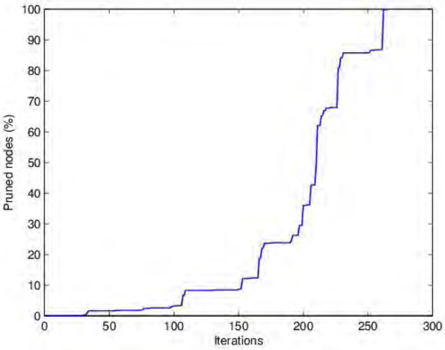

Fig. 4. Pruned nodes of BB vs. the number of iterations

IV. Numerical Results and Comparison

In this section, numerical results are presented to show the performance of the CR system under consideration with our optimal solution. The simulation setup is the same as that for generating Fig. 3 and Fig. 4.

Fig. 5 illustrates the maximum average achievable transmit rate for the SU using the BB algorithm in comparison with a reference algorithm from [6] and [7]. In this case, we assume that the average interference power is limited to 2, and the outage probability limits are set as 0.21 and 0.31, respectively. The parameter values for c in the reference algorithm [6] are set to 0.7 and 0.5, which leads to outage probabilities of 0.21 and 0.31, respectively. From this figure, we see that the transmit rate with $p_{th} = 0.31$ is always greater than or equal to the rate with $p_{th} = 0.21$, which is as expected. Moreover, we note that the maximum average achievable transmit rate with the BB approach is always higher than the rate of the reference algorithm.

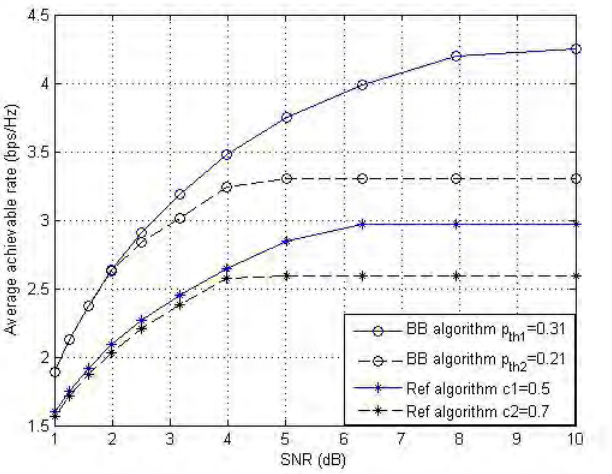

Fig. 5. Comparison of the achievable transmit rates with the BB algorithm and the reference algorithm.

V. Conclusions

In this paper, we considered a secondary communication link sharing the same spectrum with a primary link in a CR network. Multiple transmitting antennas are exploited at the SU-Tx to achieve balance between the SU transmit rate maximization and the interference regulation at the PU-Rx. We introduced the PU outage probability constraint in our formulation to model a more practical scenario, where the problem was formulated as a non-convex optimization problem with a probabilistic constraint, in addition to the SU transmit power constraint and a PU average interference constraint. To make the non-convex problem solvable, a deterministic transformation is used to approximate the original problem as an MBIP problem. An efficient BB algorithm was proposed to solve the MBIP problem, with simulation results to illustrate the superior performance of our algorithms over an existing reference algorithm.

Acknowledgment

This work is supported by the National Science & Technology Specific Program of China (No. 2010ZX03005-001-03), National High Tech. Research & Development of China (863 Program, No. 2006AA01Z272 and 2009AA02Z412) and International Science and Technology Cooperation Project (No. 2010DFA11590).

References

[1] J. Mitola, and G. Q. Maguire, "Cognitive Radio: Making Software Radios More Personal," *IEEE Pers. Commun.,* vol. 6, no. 4, pp. 13-18, Aug. 1999,

[2] S. Haykin, "Cognitive Radio: Brain-Empowered Wireless Communications," *IEEE J. Select. Areas in Commun.*, vol. 23, no. 2, Feb. 2005.

[3] M. Gastpar, "On Capacity under Receive and Spatial Spectrum-sharing Constraints," *IEEE Trans. Inf. Theory*, vol. 54, no. 2, pp. 471–487, Feb. 2005.

[4] Y. Xing, C. N. Mathur, M. A. Haleem, R. Chandramouli, and K. P. Subbalakshmi, "Dynamic Spectrum Access with QoS and Interference Temperature Constraints," *IEEE Tans. Mobile Comput.,* vol. 6, no. 4, pp. 423-433, Apr. 2007.

[5] R. Zhang, and Y.-C. Liang, "Exploiting Multi-Antennas for Opportunistic Spectrum Sharing in Cognitive Radio Networks," *IEEE J. Select. Topics in Signal Processing*, vol. 19, no. 1, pp. 88–102, Feb. 2008.

[6] L. Zhang, Y.-C. Liang, and Y. Xin, "Robust Cognitive Beamforming with Partial Channel State Information," in *Proc. Conf. Inf. Sciences and Systems (CISS)*, Mar. 2008.

[7] L. Zhang, Y.-C. Liang, and Y. Xin, "Robust Designs For MISO-Based Cognitive Radio Networks With Primary User's Partial Channel State Information," in *Proc. IEEE Global Telecomm. Conf. GLOBECOM*, Nov. 2008.

[8] J. R. Brige and F. V. Louveaux, *Introduction to Stochastic Programming*. New York: Springer, 1997.

[9] A. Ruszczynski, "Probabilistic Programming with Discrete Distributions and Precedence Constrained Knapsack Polyhedra," *Math. Program.*, vol. 93, no. 2, pp. 195–215, 2002.

[10] D. R. Morgan, J. W. Eheart, and A. J. Valocchi, "Aquifer Remediation Design Under Uncertainty Using a New Chance Constrained Programming Technique," *Water Resources Research,* vol. 29, no. 3, pp. 551-561, Mar. 1993.

[11] L. G. Mitten, "Branch-And-Bound Methods: General Formulation and Properties," *Operations Research*, Vol. 18, no. 1, pp. 24-34, 1970.

[12] S. Boyd, A. Ghosh, and A. Magnani, "Branch and Bound Methods," Notes, Stanford University, 2003.

[13] P. Beraldi and A. Ruszczynski, "A Branch and Bound Method for Stochastic Integer Problems under Probabilistic Constraints," *Optimization Methods and Software*, vol. 17, no. 3, pp. 359–382, 2002.

[14] M. Grant and S. Boyd, "CVX: Matlab Software for Disciplined Convex Programming," http://www.stanford.edu/ boyd/cvx/index.html.

[15] S. Boyd and L. Vandenberghe, *Convex Optimization.* Cambridge University Press, 2003.

[16] J. F. Sturm, "Using SeDuMi 1.02, a MATLAB Toolbox for Optimization over Symmetric Cones," Optim. Meth. Softw., vol. 11, pp. 625–653, 1999.