Convex Optimization Fall 2022 - Course project
Reading report of
Transmit Beamforming for Physical-Layer Multicasting

An Vuong
vuonga2@oregonstate.edu

Abstract

This paper focuses on the methods provided in [12], wherein the authors proposed using the Semidefinite Relaxation (SDR) based algorithms to solve the optimization problems found in transmit beamforming. This text summarizes my understanding of the aforementioned discussion, an implementation in Python is provided, and application of SDR to a similar problem called secrecy capacity is also discussed.

1 Introduction

In modern wireless communications systems, it is often the case that a single base station (BTS), being equipped with multiple antennas, will have to serve many different mobile users concurrently. If the users are well separated spatially, the bandwidth can be saved by taking advantage of the availability of multiple transmitting antennas [15]. This is the beamforming problem, in which each signal, before being transmitted to the corresponding user, is multiplied with a weight vector in order to maximize the Signal-to-Noise ratio (SNR) at the receiving users while keeping the total transmitting power within a given budget. In the case of only one user and fully-known Channel State Information (CSI), finding optimal weight vector can be done by projection onto signal subspace [3]. However, if there are multiple users, this is not a trivial task.

In [12], the authors approached the problem from a convex optimization perspective. They considered a wireless broadcast scenario where a single BTS, equipped with N antennas, transmits signals to M mobile users, each equipped with only one antenna. The problem is thus finding an optimal weight vector of N-dimensional such that the total transmit power stays within limit of P, while keeping SNR at each receiving user at an acceptable level. Different requirements of SNR can lead to different formulations:

1. Quality-of-Service (QoS)-based formulation: each receiving user is guaranteed to have a SNR that is greater than some predefined value for that user.
2. Worst case-based formulation: maximize the smallest SNR while keeping the transmitting power in check.
3. Average SNR-based formulation: this method attempts to maximize the sum of SNR of all users, this does not guarantee SNR level at each user.

The first two formulations (QoS-based and Worst case based) are the main contribution of [12], while the third one is discussed in [8]. The rest of this paper is organized as follows: next section introduces the optimization problems with respect to formulations 1 and 2 listed above, some related works on SDR are also discussed. Section 3 goes through SDR applied to optimization problems formulated in Section 2. Section 4 provides application to secrecy capacity. Section 5 tries to replicate the experimental results shown in [12], as well as the secrecy capacity problem. Section 6 concludes the paper.

2 Problem formulation and Semidefinite Relaxation

2.1 Problem formulation

Consider the scenario where a single BTS, being equipped with N antennas and transmitting power budget of P, transmits signals to M different users, each equipped with one receiving antenna. Let's denote $\mathbf{h}_i \in \mathbb{C}^N$ as channel vector corresponding to the link between BTS and user i, where $i \in \{1, 2, \dots, M\}$, $\mathbf{w} \in \mathbb{C}^N$ is beamforming vector. The following assumptions are made to simplify the derivations:

1. For now, the channel is considered to be frequency-flat, and there are only path loss and phase delay incorporated in $\mathbf{h}_i$. Different fading model is discussed later.
2. The transmitting signal x is assume to be white, with zero-mean and unit variance $\sigma_x^2 = 1$.
3. CSI is completely known at the BTS, there is no error in estimation of $\mathbf{h}_i$.
4. The BTS is in broadcasting mode, so there is no cross interference between different users. The only source of noise is modeled as receiving noise at each user's receiver, which is also assumed to be white with zero-mean and variance σ_i.

With these assumptions in mind, the transmitting power is $\|\mathbf{w}\|_2^2 \sigma_x^2 = \|\mathbf{w}\|_2^2$. The signal received at user i is $y_i = \mathbf{h}_i^H \mathbf{w} x + n_i$, where n_i is the receiving noise at user i and $(.)^H$ denotes the Hermitian conjugate. Since $\sigma_x^2 = 1$, we have the SNR at user i: $\text{SNR}_i = \frac{|\mathbf{h}_i^H \mathbf{w}|^2}{\sigma_i^2}$.

In the QoS-based formulation, we want to guarantee a lower bound for each SNR_i, let us denote this lower bound as $\rho_{min,i}$, corresponding to the smallest acceptable SNR level at user i. The problem is now to minimize $\|\mathbf{w}\|_2^2$ while keeping $\text{SNR}_i \geq \rho_{min,i}$:

$$\begin{aligned} \min_{\mathbf{w} \in \mathbb{C}^N} \quad & \|\mathbf{w}\|_2^2 \\ \text{s.t.} \quad & \frac{|\mathbf{h}_i^H \mathbf{w}|^2}{\sigma_i^2} \geq \rho_{min,i}, \ i \in \{1, 2, \dots, M\} \end{aligned} \tag{1}$$

We have ignored the power constraint P for the QoS formulation, since this constraint might make the problem infeasible if $\rho_{min,i}$ is set too high. If we now normalize the channel vector $\tilde{\mathbf{h}_i} = \mathbf{h}_i / \sqrt{\sigma_i^2 \rho_{min,i}}$, problem (1) can be written in a simpler form:

$$\begin{aligned} \mathcal{Q}: \qquad \min_{\mathbf{w} \in \mathbb{C}^N} \quad & \|\mathbf{w}\|_2^2 \\ \text{s.t.} \quad & |\tilde{\mathbf{h}_i}^H \mathbf{w}|^2 \geq 1, \ i \in \{1, 2, \dots, M\} \end{aligned} \tag{2}$$

Continue on to the formulation based on worst case SNR. In this scenario, we want to maximize the smallest SNR_i of user i while adhering to the constraint on the transmitting power. This can be formulated as:

$$\begin{aligned} \max_{\mathbf{w} \in \mathbb{C}^N} \min_i \quad & \left\{ \frac{|\mathbf{h}_i^H \mathbf{w}|^2}{\sigma_i^2} \right\} \\ \text{s.t.} \quad & \|\mathbf{w}\|_2^2 \leq P \end{aligned} \tag{3}$$

Similar to the QoS case, we can normalize the channel vector and get to the following problem:

$$\begin{aligned} \mathcal{F}: \qquad \max_{\tilde{\mathbf{w}} \in \mathbb{C}^N} \min_i \quad & \left\{ |\mathbf{h}_i'^H \tilde{\mathbf{w}}|^2 \right\} \\ \text{s.t.} \quad & \|\tilde{\mathbf{w}}\|_2^2 \leq 1 \end{aligned} \tag{4}$$

Since P is just a constant, it can be ignored from optimization perspective by normalizing $\mathbf{w}$: $\tilde{\mathbf{w}} = \mathbf{w}/\sqrt{P}$, the resulting optimal objective value will just need to be scaled by P. It is noted that in the special case where all $\rho_{min,i}$'s are the same and are equal to ρ_{min}, and P is set to P_q, which is the minimum transmitting power obtained from solving $\mathcal{Q}$, then problem $\mathcal{Q}$ and $\mathcal{F}$ are equivalent up to a scaling factor since their solutions also satisfy each other's constraints. This observation is important since in the appendix of [12], $\mathcal{Q}$ is shown to be NP-hard by mapping it into the well-known partition problem [5], now we also have $\mathcal{F}$ is equivalent to $\mathcal{Q}$ in a special case, this makes $\mathcal{F}$ an NP-hard problem as well.

To construct a general framework for solving this type of problems, some forms of simplification and approximation must be employed in order to run within polynomial time. One such approximation is Semidefinite Relaxation, which will be briefly reviewed in the next subsection.

2.2 Semidefinite Relaxation

Consider the following quadratically constrained quadratic program (QCQP) problem:

$$\begin{aligned} \mathcal{V}: \qquad \min_{\mathbf{x}\in\mathbb{R}^n} \quad & \mathbf{x}^T\mathbf{A}\mathbf{x} \\ \text{s.t.} \quad & \mathbf{x}^T\mathbf{F}_i\mathbf{x} \geq f_i \quad i = 1, 2, \dots, M \end{aligned} \tag{5}$$

where $\mathbf{A}$ and $\mathbf{F}_i$ are positive semidefinite (PSD) matrices. This problem is known to be NP-hard [14], since the feasible set of $\mathcal{V}$ is an intersection of multiple ellipsoids, this makes the problem difficult to solve efficiently.

The relaxation for $\mathcal{V}$ is based on the fact that given a matrix $\mathbf{C} \in \mathbb{S}^n$, we have:

$$\mathbf{x}^T\mathbf{C}\mathbf{x} = \text{trace}(\mathbf{x}^T\mathbf{C}\mathbf{x}) = \text{trace}(\mathbf{C}\mathbf{x}\mathbf{x}^T) \tag{6}$$

which is linear in $\mathbf{x}\mathbf{x}^T$, let us denote matrix $\mathbf{X} = \mathbf{x}\mathbf{x}^T$, it is obvious that $\mathbf{X}$ is PSD and of rank-1. Using this, 7 can be rewritten as:

$$\begin{aligned} \min_{\mathbf{X}\in\mathbb{S}^n} \quad & \text{trace}(\mathbf{A}\mathbf{X}) \\ \text{s.t.} \quad & \text{trace}(\mathbf{F}_i\mathbf{X}) \geq f_i \quad i = 1, 2, \dots, M \\ & \mathbf{X} \succeq 0 \\ & \text{rank}(\mathbf{X}) = 1 \end{aligned} \tag{7}$$

Since everything in 7 is convex except for the rank-1 requirement, if we drop this constraint, this leads to the semidefinite relaxation:

$$\begin{aligned} \mathcal{V}_{\text{SDR}}: \qquad \min_{\mathbf{X}\in\mathbb{S}^n} \quad & \text{trace}(\mathbf{A}\mathbf{X}) \\ \text{s.t.} \quad & \text{trace}(\mathbf{F}_i\mathbf{X}) \geq f_i \quad i = 1, 2, \dots, M \\ & \mathbf{X} \succeq 0 \end{aligned} \tag{8}$$

At first it can be somewhat off-putting to just drop a constraint, but in practice SDR has been shown to work surprisingly well in multiple different problems. For e.g., [4] applies SDR to the famous Maximum Cut problem, which is one of the 21 NP-hard problems from Karp's list [6], in this setting, the SDR solution is shown to be achieving at least 0.87856 of the optimal value. In [13], SDR is applied for multiple users detection within CDMA setting, it produces solutions which closely match with that of a maximum likelihood detector, the authors also show that SDR formulation has faster computation time as the number of users grows.

These achievements all rely on a crucial step: recover $\mathbf{x}_{\text{opt}}$ from $\mathbf{X}_{\text{opt}}^{\text{SDR}}$, where $\mathbf{x}_{\text{opt}}$ and $\mathbf{X}_{\text{opt}}^{\text{SDR}}$ denote the optimal solutions of $\mathcal{V}$ and $\mathcal{V}_{\text{SDR}}$ respectively. In general, $\mathbf{X}_{\text{opt}}^{\text{SDR}}$ is not rank-1, and thus is different from $\mathbf{x}_{\text{opt}}\mathbf{x}_{\text{opt}}^T$. The crutch of SDR is thus how to recover $\mathbf{x}_{\text{opt}}$ from $\mathbf{X}_{\text{opt}}^{\text{SDR}}$ efficiently. This recovery procedure is of course not perfect, since otherwise we would have solved an NP-hard problem efficiently. Section 3.2 introduces a few commonly used methods for this operation. It is noted that there are cases where $\mathbf{X}_{\text{opt}}^{\text{SDR}}$ is exactly the same as $\mathbf{x}_{\text{opt}}\mathbf{x}_{\text{opt}}^T$, which means SDR solves the problem optimally. An example can be found in [18], wherein the authors show that if every off-diagonal entries of $\mathbf{A}$ is non-negative, then SDR gives optimal solution within polynomial time, and $\mathbf{x}_{\text{opt}} = \sqrt{\text{diag}(\mathbf{X}_{\text{opt}}^{\text{SDR}})}$.

2.3 Relation to dual problem

Consider problem $\mathcal{V}$ (5), the Lagrangian of $\mathcal{V}$ is:

$$\mathcal{L}(\mathbf{x}, \lambda) = \mathbf{x}^T\mathbf{x} + \sum_{i=1}^{M}\lambda_i(1 - \mathbf{x}^T\mathbf{F}_i\mathbf{x}) = \mathbf{x}^T(\mathbf{I} - \sum_{i=1}^{M}\lambda_i\mathbf{F}_i)\mathbf{x} + \sum_{i=1}^{M}\lambda_i$$

and the dual problem is:

$$\mathcal{V}_{\text{dual}}: \qquad \max_{\lambda \succeq 0} \min_{\mathbf{x}} \ \mathbf{x}^T(\mathbf{I} - \sum_{i=1}^{M} \lambda_i \mathbf{F}_i)\mathbf{x} + \sum_{i=1}^{M} \lambda_i \tag{9}$$

[16] shows that (9) is equivalent to the SDR formulation (8). This can be shown by noting that minimization over $\mathbf{x}$ in $\mathcal{V}_{\text{dual}}$ is attainable only if $\mathbf{I} - \sum_{i=1}^{M} \lambda_i \mathbf{F}_i$ is PSD, otherwise its minimum would be negative infinity. Assuming that $\mathbf{I} - \sum_{i=1}^{M} \lambda_i \mathbf{F}_i$ is PSD, minimization over $\mathbf{x}$ is attained when $\mathbf{x} = \mathbf{0}$, the dual problem now reduces to a semidefinite program:

$$\begin{aligned} &\max_{\lambda \succeq 0} \ \sum_{i=1}^{M} \lambda_i \\ &\text{s.t.} \ \ \mathbf{I} - \sum_{i=1}^{M} \lambda_i \mathbf{F}_i \succeq 0 \ \ i = 1, 2, \ldots, M \end{aligned} \tag{10}$$

In [12], the authors show that one can use (10) to solve for λ and then perform optimization over $\mathbf{x}$ with such λ. This should provide a lower bound for $\mathcal{V}$ [1], this bound is then can be compared with the solution given by SDR for evaluation. Section 5 uses similar idea for MaxMin formulation. It is noted that the above analysis also applies to complex vector, since any complex $\mathbf{w} \in \mathbb{C}^N$ can always be represented as a real-valued $\mathbf{x} \in \mathbb{R}^{2N}$, similarly for matrix.

3 Application to Transmit beamforming problem

3.1 Relaxation

Applying SDR, problem $\mathcal{Q}$ (2) can be recast as:

$$\begin{aligned} \mathcal{Q}_{\text{SDR}}: \qquad \min_{\mathbf{W} \in \mathbb{C}^{N \times N}} \ & \text{trace}(\mathbf{W}) \\ \text{s.t.} \ & \text{trace}(\mathbf{W}\tilde{\mathbf{H}}_i) \geq 1, \ i \in \{1, 2, \ldots, M\} \\ & \mathbf{W} \succeq 0 \end{aligned} \tag{11}$$

where $\mathbf{W} = \mathbf{w}\mathbf{w}^H$, which is an $N \times N$ complex and PSD matrix, similarly $\tilde{\mathbf{H}}_i = \tilde{\mathbf{h}}\tilde{\mathbf{h}}^H$. It is noted that $\mathbf{W}$ is of rank-1 but this constraint has been dropped as a relaxation. In [12], the authors use SeDuMi, which is a Matlab interface for CVX package, this requires (11) to be expressed in standard form, which they achieve by vectorizing $\mathbf{W}$, $\tilde{\mathbf{H}}$ combining with the introduction of some slack variables. Fortunately CVXPY[1] does not have this requirement, which makes (11) good enough.

With a similar fashion, problem $\mathcal{F}$ (4) can be rewritten as:

$$\begin{aligned} \mathcal{F}'_{\text{SDR}}: \qquad \max_{\mathbf{W} \in \mathbb{C}^{N \times N}} \min_{i} \ & \text{trace}(\mathbf{W}\mathbf{H}'_i) \\ \text{s.t.} \ & \mathbf{W} \succeq 0, \ \ \text{trace}(\mathbf{W}) \leq P \end{aligned} \tag{12}$$

where $\mathbf{H}'_i = \mathbf{h}'\mathbf{h}'^H$. It is noted again that $\tilde{\mathbf{h}_i} = \mathbf{h}_i / \sqrt{\sigma_i^2 \rho_{min,i}}$ while $\mathbf{h}'_i = \mathbf{h}_i / \sigma_i$. In order to solve (12), a new variable t is introduced:

$$\begin{aligned} \mathcal{F}_{\text{SDR}}: \qquad \max_{\mathbf{W} \in \mathbb{C}^{N \times N}, \, t \in \mathbb{R}} \ & t \\ \text{s.t.} \ & \text{trace}(\mathbf{W}\mathbf{H}'_i) \geq t, \ i \in \{1, 2, \ldots, M\} \\ & \mathbf{W} \succeq 0, \ \ \text{trace}(\mathbf{W}) \leq P \end{aligned} \tag{13}$$

again (13) can be input directly into CVXPY, so conversion to standard form is not needed. Solving $\mathcal{Q}_{\text{SDR}}$ or $\mathcal{F}_{\text{SDR}}$ in CVXPY would give us optimal $\mathbf{W}_{\text{opt}}$. As discussed in Section 2.2, we now need to recover the beamforming vector $\mathbf{w}_{\text{opt}}$ from $\mathbf{W}_{\text{opt}}$, next section introduces several methods to achieve this.

[1] `https://www.cvxpy.org`

3.2 Recover $\mathbf{w}_{\text{opt}}$ from $\mathbf{W}_{\text{opt}}$

Since $\mathbf{W}_{\text{opt}}$ is a square matrix, it has an eigendecomposition $\mathbf{W}_{\text{opt}} = \mathbf{U\Sigma U}^H$. We have $\mathbf{W}_{\text{opt}}$ is also PSD, so all of its eigenvalues are non-negative, one can simply pick $\mathbf{w}_{\text{opt}}$ as the eigenvector $\mathbf{u}_1$ corresponding to the largest eigenvalue. This works well if $\mathbf{W}_{\text{opt}}$ has only one dominant eigenvalue, i.e all other eigenvalues are much smaller than the first one. In general, such might not be the case, techniques based on randomization were introduced to overcome this.

In [10], based on Cholesky factorization $\mathbf{W}_{\text{opt}} = \mathbf{V}^T\mathbf{V}$, the authors approximate $\mathbf{w}_{\text{opt}}$ as:

$$\mathbf{w}_{\text{opt}} = h(\mathbf{V}^T\mathbf{e}) \tag{14}$$

where $\mathbf{e}[k]$ is independent random complex number uniformly distributed on a unit circle, and $h(\mathbf{x}) : \mathbb{R}^n \longrightarrow \mathbb{R}^n$ is a function such that $h(\mathbf{x}) = 1$ if $\mathbf{x}[k] \geq 0$ and -1 otherwise. Based on this idea, [12] introduced the first randomization method:

$$\text{randA}: \quad \mathbf{w}_{\text{opt}} = \mathbf{U\Sigma}^{1/2}\mathbf{e} \tag{15}$$

Recall from Section 2.2, where [14] shows that SDR, in some special cases, can give optimal solution and the recovered $\mathbf{w}_{\text{opt}} = \sqrt{\text{diag}(\mathbf{W}_{\text{opt}})}$. This motivates another randomization approach:

$$\text{randB}: \quad \mathbf{w}_{\text{opt}}[k] = \sqrt{\mathbf{W}_{\text{opt}}[k,k]}\mathbf{e}[k] \tag{16}$$

where $\mathbf{x}[k]$ denotes k-th element of vector $\mathbf{x}$ and $\mathbf{X}[k,k]$ denotes element of $\mathbf{X}$ found at row k and column k. The vector $\mathbf{e}$ can be construct by generating $\mathbf{e}[k] = e^{j\theta_k}$ where $\theta_k \sim \text{Uniform}[0, 2\pi)$.

One can also draw $\mathbf{e}[k]$ from a complex circularly symmetric normal distribution. Let $\mathbf{n}$ denote this random vector to avoid confusion with previous methods, we have $\mathbf{n}[k] \sim \mathcal{CN}(0,1)$:

$$\text{randC}: \quad \mathbf{w}_{\text{opt}} = \mathbf{U\Sigma}^{1/2}\mathbf{n} \tag{17}$$

this procedure makes $\mathbf{w}_{\text{opt}}\mathbf{w}_{\text{opt}}^T = \mathbf{W}_{\text{opt}}$ in expectation.

It should be noted that $\mathbf{w}_{\text{opt}}$ generated in these cases might not satisfy the constraint $|\tilde{\mathbf{h}}_i^H\mathbf{w}|^2 \geq 1$, thus one could scale $\mathbf{w}_{\text{opt}}$ by scalar value such that all constraints are met. [12] uses this as a way to evaluate the effectiveness of each randomization method, i.e the best method should require the smallest scaling factor. A metric, namely upper bound on power boost (UBPB), is then defined as $\|\mathbf{w}^*_{\text{opt}}\|_2^2/\text{trace}(\mathbf{W}_{\text{opt}})$, where $\mathbf{w}^*_{\text{opt}}$ is the $\mathbf{w}_{\text{opt}}$ that needs smallest scaling. This also concludes the section, next part will introduce the secrecy capacity problem and its SDR formulation.

4 Secrecy capacity problem

4.1 Problem formulation

Consider a scenario where a BTS Alice is wirelessly transmitting signal to a legitimate receiver Bob. We know that in some vicinity around Bob, there is an illegitimate eavesdropper Eve, who will try to sniff information from the link between Alice and Bob by passively listening to the radiated signal from the Alice. From information theory perspective, such scenario can be characterized by the difference in channel capacities between Alice and Bob and between Alice and Eve [17] [7]:

$$C_S = C(Alice; Bob) - C(Alice; Eve) \tag{18}$$

where $C(A;B)$ denotes channel capacity between A and B, C_S denotes the secrecy capacity. Alice can transmit information to Bob securely without leaking any information to Eve if the secrecy rate $C_S > 0$ [11]. For AWGN channel we have:

$$C_S = \log(1 + \text{SNR}_B) - \log(1 + \text{SNR}_E) \tag{19}$$

where SNR_B, SNR_E are signal-to-noise ratio at Bob and Eve respectively.

Assuming Alice is equipped with N antennas, Bob and Eve each only has one antenna. The signals received at Bob and Eve can be written as:

$$y_B = \mathbf{h}^T\mathbf{w}x + n_B \tag{20}$$

$$y_E = \mathbf{g}^T\mathbf{w}x + n_E \tag{21}$$

where x is the transmitted signal with zero-mean and unit-variance, y_B and y_E are signals received at Bob and Eve, $n_B \sim \mathcal{CN}(0, \sigma_B^2)$ and $n_E \sim \mathcal{CN}(0, \sigma_E^2)$ are noises at Bob and Eve since the channels are assumed to be AWGN. $\mathbf{h} \in \mathbb{C}^N$, $\mathbf{g} \in \mathbb{C}^N$ are channel vectors from Alice to Bob and Eve, respectively. $\mathbf{w} \in \mathbb{C}^N$ is beamforming vector, similar to Section 2.1. From these assumptions, we have:

$$C_S = \log(1 + \frac{|\mathbf{h}^T \mathbf{w}|^2}{\sigma_B^2}) - \log(1 + \frac{|\mathbf{g}^T \mathbf{w}|^2}{\sigma_E^2}) \tag{22}$$

From optimization perspective, one wants to design the beamforming vector $\mathbf{w}$ such that C_s is maximized. To simplify the problem, let us assume CSI is perfectly known at Alice, i.e no estimation error on $\mathbf{h}$ and $\mathbf{g}$, this leads to the following optimization problem:

$$\begin{aligned} \mathcal{C}: \qquad & \max_{\mathbf{w}} \quad \log(1 + \frac{|\mathbf{h}^T \mathbf{w}|^2}{\sigma_B^2}) - \log(1 + \frac{|\mathbf{g}^T \mathbf{w}|^2}{\sigma_E^2}) \\ & \text{s.t} \quad \|\mathbf{w}\|^2 \leq P \end{aligned} \tag{23}$$

4.2 Relaxation

One can apply SDR to (23) and get to the following formulation:

$$\begin{aligned} & \max_{\mathbf{W}} \quad \log(1 + \frac{\mathbf{h}^T \mathbf{W} \mathbf{h}}{\sigma_B^2}) - \log(1 + \frac{\mathbf{g}^T \mathbf{W} \mathbf{g}}{\sigma_E^2}) \\ & \text{s.t} \quad \mathbf{W} \succeq 0, \quad \text{Tr}(\mathbf{W}) \leq P \end{aligned} \tag{24}$$

where $\mathbf{W} = \mathbf{w}\mathbf{w}^H$ and the rank-1 constraint has been dropped. Since the objective function of (24) is neither convex nor concave, let $s = 1 + \frac{\mathbf{g}^T \mathbf{W} \mathbf{g}}{\sigma_E^2}$, (24) becomes:

$$\begin{aligned} \mathcal{C}_{\text{SDR}}: \qquad & \max_{\mathbf{W}} \quad \log(1 + \frac{\mathbf{h}^T \mathbf{W} \mathbf{h}}{\sigma_B^2}) - \log(s) \\ & \text{s.t} \quad \mathbf{W} \succeq 0, \quad \text{Tr}(\mathbf{W}) \leq P \\ & \qquad s = 1 + \frac{\mathbf{g}^T \mathbf{W} \mathbf{g}}{\sigma_E^2} \end{aligned} \tag{25}$$

By iterate through multiple different values of s, we can find the pair $\{s_{\text{opt}},\ \mathbf{W}_{\text{opt}}\}$ that maximizes C_S using CVXPY. Then one can apply Section 3.2 to get back $\mathbf{w}_{\text{opt}}$.

5 Simulation results

All codes are available at `https://github.com/anvuongb/ece569_cvxopt_final`

5.1 Transmit beamforming

This section attempts to replicate the experiments provided in [12]. To generate channel vectors, location (angle and distance from the BTS) of each user is randomly generated: $\text{angle} = \text{Uniform}[0, 2\pi)$, $\text{distance} = \text{Uniform}[1000, 2500)$, units are radians and meters respectively. Channel coefficient is assumed to have Circularly Symmetric Complex Gaussian (CSCG) distribution, and antennas separation is $\lambda_c/2$, where λ_c is carrier wavelength, this is typical in multiple antennas system. This simply multiples the k-th channel coefficient by baseband gain $e^{-j2\pi d_i/\lambda_c}$, where d_i is distance from BTS to user i. $P = 1$, σ_i, $\rho_{min,i}\sigma_i^2$ is set to 1 for all i, note that this makes MaxMin and QoS formulations equivalent up to a scalar, as discussed in Secton 2.2. Table 1 and 2 show UBPB for different combinations of N and M, they match well with that of [12], approximation results also seem to get better as number of randomization increase.

As a sanity check, Fig.1a and 1b shows the beam pattern (N=8, M=16) for line-of-sight (LoS) and multipath fading channels, solved by QoS & MaxMin SDR. We can see that in LoS case (Fig.1a), the beam power is directly proportional to the distance, which makes sense. With the fading channels (Fig.1b), the beam powers are different compared to those in LoS case. This makes sense since the channels' gains are from a random process. It is also apparent that MaxMin solution is also QoS solution scaled by some factor, since we have specifically chosen $\sigma_i = 1$ and $\rho_{min,i}\sigma_i^2 = 1$, recall Section.2.1. This effect is less obvious in Fig.1b since we have ignored the baseband gain in multipath simulations.

To make comparison between QoS and MaxMin, t in $\mathcal{F}_{\text{SDR}}$ is considered the relaxation bound (highest possible $\min_{\text{SNR}_i}$). $\mathbf{w}_{\text{opt}}$ obtained from solving $\mathcal{F}_{\text{SDR}}$ & $\mathcal{Q}_{\text{SDR}}$ (with all three randomization techniques) are scaled to norm P and used to compute the corresponding $\min_{\text{SNR}_{i,\text{rand}}}$. The percentage of difference between this value and the relaxation bound is shown in Fig.1c, note that channels coefficients are generated with larger mean and variance in this case to make the visual easier to see. It can be seen again that both QoS & MaxMin formulation gives the same solution on average (around 30% less than the relaxation bound).

Table.3 further shows the evaluations of $\min_{\text{SNR}_i}$ from different methods, the optimal t in (13), found by CVXPY[2], is considered as upper bound for $\min_{\text{SNR}_i}$ thanks to Lagrangian duality, as briefly mentioned in Section.2.3. After randomization, $\mathbf{w}_{\text{opt}}$ is used to calculate the $\min_{\text{SNR}_i}$, which is shown in MaxMin column. The $\min_{\text{SNR}_i}$ found by maximizing average SNR [9] is given in MaxAverage column. As a sanity check, No BMF column shows SNR when no beamforming is used, this is achieved by using $\mathbf{w} = 1/\sqrt{N}\mathbf{1}^T$. The data is averaged over 1000 Monte Carlo runs. In each run, the channel vector $\mathbf{h}_i$ is redrawn from a CSCG distribution, and $30 \times N \times M$ randomizations are performed for each technique (randA, randB, randC). It can be seen that MaxMin SDR with randomizations provides solutions which are surprisingly close to the upper bound. While MaxAvergave gives worse results in term of SNR, it should be noted that its computational complexity is much lower than that of MaxMin, since it simply maximizes over the average.

This concludes the simulations for transmit beamforming[3], they almost match exactly to those of [12]. Next section demonstrate the application to secrecy capacity.

N/M	mean	std
4/8	1.0732	0.2796
4/16	1.2809	0.4888
8/16	1.4619	0.5002
8/32	2.3915	0.8346

Table 1: QoS UBPB MC - 1000 randomization

N/M	mean	std
4/8	1.0792	0.2880
4/16	1.2898	0.4853
8/16	1.4567	0.5043
8/32	2.3815	0.8529

Table 2: QoS UBPB MC - 30xNxM randomization

N/M	upper bound	MaxMin	MaxAverage	No BMF
4/8	1.0248	0.9185	0.2424	0.1250
4/16	0.7367	0.5185	0.1090	0.0660
8/16	1.4462	0.8531	0.1461	0.0613
8/32	1.0777	0.4420	0.0587	0.0320

Table 3: Comparison between MaxMin and MaxAverage

5.2 Secrecy capacity

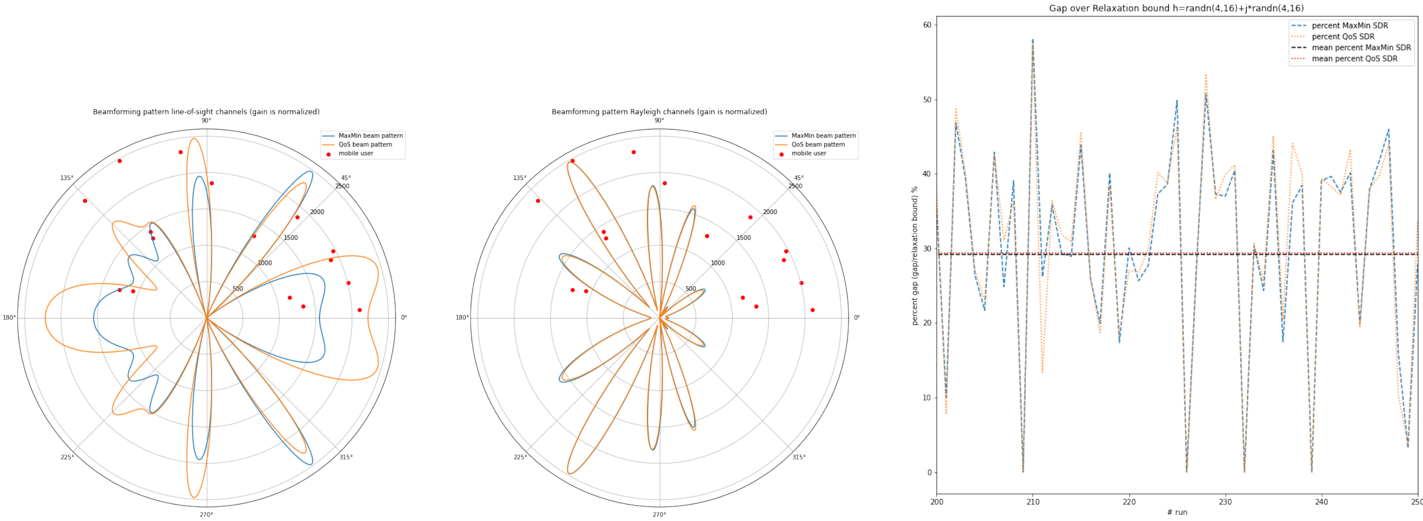
Consider problem $\mathcal{C}_{\text{SDR}}$ (25), parameters P, σ_B, σ_E are all set to 1. Locations of Bob and Eve are set as (1202m, 3.0717rad) and (2057m, 0.5235rad) respectively. Fig.2 shows the simulation in LoS channels case, all three randomization techniques are used (with $30 \times N \times M$ runs), $\mathbf{w}_{\text{opt}}$ are scaled to norm P, the vector achieving best secrecy capacity C_S is then chosen. We can see the beamforming pattern is generated in such a way to minimize signal received as Eve, as expected. Fig.2b shows C_S as a function of s, which is the scalar we use to augment the optimization problem, it looks like a convex function w.r.t s and it is in fact convex if some constraints are satisfied [2]. It is noted that $N = 8$ antennas are used for Fig.2a and 2b.

To investigate advantages of having multiple antennas, Fig.2c visualizes C_S as a function of distance d between Eve and Bob. In this case, starting from the position as shown in Fig.2, Eve then gets closer to Bob by 100 meters at each time step and C_S is plotted. It can be seen that having multiple antennas lead to much better secrecy capacity overall. An animation for beamforming pattern when Eve is moving can be found at `https://tinyurl.com/secbeam`.

Table.4 compares secrecy capacity obtained by different randomization methods. The secrecy capacity achieved after randomization and scaling are averaged over 300 runs, channels vectors are redrawn

[2]MOSEK solver was used in all experiments since it was more robust, a free license is available from `www.mosek.com`

[3]Experiments with VDSL channel data were omitted since I could not find the data online.



(a) Line-of-sight channels

(b) Multipath channels

(c) Comparison between QoS & MaxMin

Figure 1: Beamforming pattern (N=8, M=16) and gap between QoS & MaxMin solutions

from CSCG each run. It can be seen that randA and randC perform very similarly, while randB consistently performs worse than the others.

6 Conclusion

This paper has taken a look at [12], where Semidefinite Relaxation has been shown to be a simple yet powerful technique to solve NP-hard problems related to beamforming. Multiple randomization techniques are introduced to recover the optimal vector from a PSD matrix. They provides surprisingly good results and in many cases can closely match that of the optimal bound. Almost all experiments (except for VDSL) are successfully replicated after some painful mistakes made in generating the channels vectors, I found that

For Secrecy capacity problem, even though it is a bit more complicated since the objective function is neither concave nor convex, it still works well. By looking at the beamforming pattern, we can see that SDR tries its best to place Eve in the null channel of the transmitting signal, which makes sense. Since I'm interested in the Secrecy capacity problem, this has been a great tool to learn. I'm wondering if there are some convex functions that can approximate the objective of $\mathcal{C}_{\mathrm{SDR}}$ (25) well. This would make solving it a lot more robust, since sometimes the scalar s can lead to numerical instability as I have encountered several times using CVXPY.

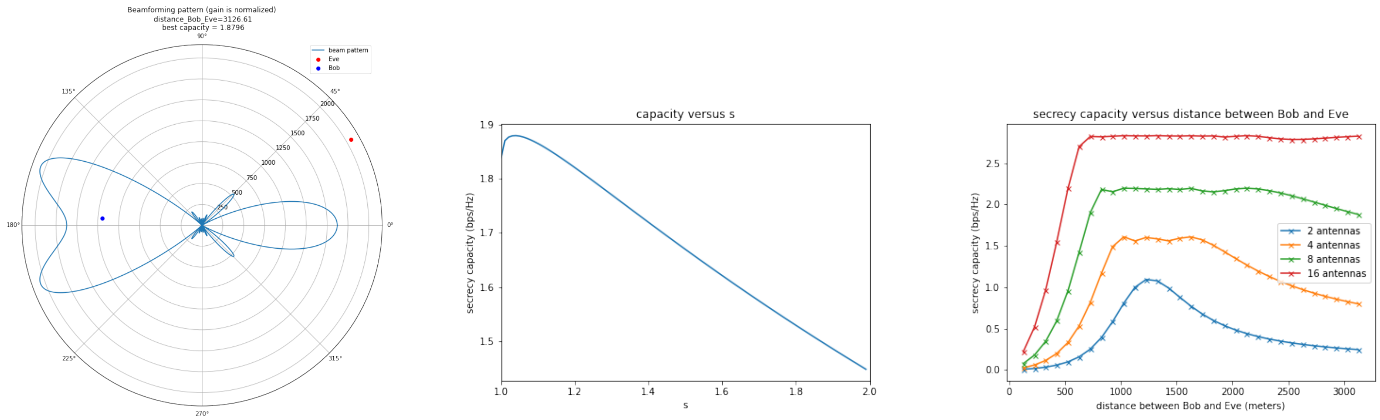
This also finalizes this paper. Thank you for a very informative course.

N	randA	randB	randC
2	0.7597	0.7593	0.7597
4	1.4079	1.3674	1.4079
8	1.9968	1.7226	1.9969
16	2.7448	2.1286	2.7449

Table 4: Secrecy comparison between different randomization techniques

References

[1] Boyd, S., Boyd, S. P., and Vandenberghe, L. *Convex optimization*. Cambridge university press, 2004.



(a) Best C_S beamforming pattern (LoS)

(b) C_S versus s

(c) C_S versus distance between Eve and Bob

Figure 2: Secrecy capacity

[2] CHAKRAVARTY, J., JOHNSON, O., AND PIECHOCKI, R. A convex scheme for the secrecy capacity of a mimo wiretap channel with a single antenna eavesdropper. In *ICC 2019-2019 IEEE International Conference on Communications (ICC)* (2019), IEEE, pp. 1–5.

[3] FELDMAN, D., AND GRIFFITHS, L. A projection approach for robust adaptive beamforming. *IEEE Transactions on Signal Processing 42*, 4 (1994), 867–876.

[4] GOEMANS, M. X., AND WILLIAMSON, D. P. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM (JACM) 42*, 6 (1995), 1115–1145.

[5] HARTMANIS, J. Computers and intractability: a guide to the theory of np-completeness (michael r. garey and david s. johnson). *Siam Review 24*, 1 (1982), 90.

[6] KARP, R. M. Reducibility among combinatorial problems. In *Complexity of computer computations*. Springer, 1972, pp. 85–103.

[7] LEUNG-YAN-CHEONG, S., AND HELLMAN, M. The gaussian wire-tap channel. *IEEE transactions on information theory 24*, 4 (1978), 451–456.

[8] LOPEZ, M. J. *Multiplexing, scheduling, and multicasting strategies for antenna arrays in wireless networks*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 2002.

[9] LOPEZ, M. J. Multiplexing, scheduling, and multicasting strategies for antenna arrays in wireless networks. Tech. rep., MASSACHUSETTS INST OF TECH CAMBRIDGE DEPT OF ELECTRICAL ENGINEERING AND ..., 2002.

[10] MA, W.-K., DAVIDSON, T. N., WONG, K. M., LUO, Z.-Q., AND CHING, P.-C. Quasi-maximum-likelihood multiuser detection using semi-definite relaxation with application to synchronous cdma. *IEEE transactions on signal processing 50*, 4 (2002), 912–922.

[11] SHANNON, C. E. A mathematical theory of communication. *The Bell system technical journal 27*, 3 (1948), 379–423.

[12] SIDIROPOULOS, N. D., DAVIDSON, T. N., AND LUO, Z.-Q. Transmit beamforming for physical-layer multicasting. *IEEE transactions on signal processing 54*, 6 (2006), 2239–2251.

[13] TAN, P. H., AND RASMUSSEN, L. K. The application of semidefinite programming for detection in cdma. *IEEE journal on selected areas in communications 19*, 8 (2001), 1442–1449.

[14] TSENG, P. Further results on approximating nonconvex quadratic optimization by semidefinite programming relaxation. *SIAM Journal on Optimization 14*, 1 (2003), 268–283.

[15] VAN VEEN, B., AND BUCKLEY, K. Beamforming: a versatile approach to spatial filtering. *IEEE ASSP Magazine 5*, 2 (1988), 4–24.

[16] WOLKOWICZ, H., SAIGAL, R., AND VANDENBERGHE, L. *Handbook of semidefinite programming: theory, algorithms, and applications*, vol. 27. Springer Science & Business Media, 2012.

[17] WYNER, A. D. The wire-tap channel. *Bell system technical journal 54*, 8 (1975), 1355–1387.

[18] ZHANG, S. Quadratic maximization and semidefinite relaxation. *Mathematical Programming 87*, 3 (2000), 453–465.